July 5, 1984

DCIEM REPORT NO. 84-R-22

# THE INTERACTIVE SPATIAL INFORMATION SYSTEM

M. M. Taylor
C. A. McCann
M. I. Tuori

Defence and Civil Institute of Environmental Medicine
1133 Sheppard Avenue West, P.O. Box 2000
Downsview, Ontario M3M 3B9

DEPARTMENT OF NATIONAL DEFENCE - CANADA

## Table of Contents


The Interactive Spatial Information System ..... 1
Abstract ..... 1

The Interactive Spatial Information System ..... 2
Introduction: The ISIS project ..... 1
Space and Language ..... 1
ISIS: The Vision ..... 3
What should a full ISIS do? ..... 3
Decisions and assistants ..... 3
Some required capabilities ..... 5
Some conclusions about ISIS ..... 7
Insecure information ..... 7
ISIS as a communications medium ..... 8
Spatial thought and spatial display ..... 8
Spatial display of non-spatial data ..... 9
The Vision: Conclusion ..... 10
SDBMS-1: A demonstration testbed ..... 10
MASCOT technology ..... 11
The MASCOT structure of SDBMS-1 ..... 12
Data structure in SDBMS-1 ..... 13
Multimodal interaction in SDBMS-1 ..... 15
SDBMS-1: Summary ..... 16
The Road from SDMBS-1 to ISIS ..... 17
The structure of interaction ..... 17
Protocol as syntax ..... 18
Layered protocols and multimodal interaction ..... 20
Functional elements: scoping and modification ..... 21
Spatial data representation ..... 23
Information zoom ..... 23
Browsing and information zoom ..... 24
Active Objects ..... 27
An intermediate ISIS ..... 27
Conclusions ..... 28
Figure Captions ..... 30
References ..... 32

# THE INTERACTIVE SPATIAL INFORMATION SYSTEM

M.M.Taylor, C.A.McCann and M.I.Tuori
Defence and Civil Institute of Environmental Medicine
Box 2000, Downsview, Ontario, Canada M3M 3B9

## *ABSTRACT*

An interactive spatial information system (ISIS) offers novel opportunities for decision aiding. Its synoptic and dynamic information can assist the user to discover as well as to evaluate options. Spatial display may help the symbiosis between human intuition and computer logic. The ISIS should be able to perform intelligent operations on its data, detecting inconsistencies between new and old information, assisting in the evaluation of plans, and so forth. DCIEM has developed a miniature ISIS called SDBMS-1, using an extension of a commercial relational database management system. It has been implemented using MASCOT technology, to permit easy modification as a consequence of experience. It allows the user to interact by keyboard, voice, or graphic gesture, and provides output alphanumerically or graphically. The information consists of topographic data from digitized standard maps and synthetic tactical data representing military operations. Development of an improved ISIS requires analysis of the nature of the Human-Computer dialogue. Some thoughts are presented, and an illustration given of interaction through "Active Objects" on the graphic display.

# THE INTERACTIVE SPATIAL INFORMATION SYSTEM

## *INTRODUCTION: THE ISIS PROJECT*

The ISIS (Interactive Spatial Information System) project has a double purpose: to study the interaction of humans with spatial data for decision aiding, and to provide models on which operational spatial information systems can be based. The models take the form of working systems, whose design derives from the conceptual understanding achieved at some particular stage. The primary application area is military command and control, though the potential applications cover a much wider field.

The methods of ISIS depend less on formal experiment than on iterative design and on studies of the literature in cognitive science, computer science and software engineering. Cognitive science provides insight into human use of data, and into the prospects for using artificial intelligence, and into the nature of dialogues. Computer science permits the design of effective data structures and the building of an information environment within which experiments on computer-human interaction can be performed. Software engineering studies lead to the development of methods for flexible design of multiprocessor systems that can change to meet changing understanding of computer-human dialogue.

This paper is divided into three major sections. The first section discusses the nature and uses of spatial information, and considers the functions of an idealized ISIS. The second reports work on an actual spatial information system (SDBMS-1), built around an extended version of a commercial relational database management system. SDMBS-1 is being implemented using the MASCOT technology, so as to permit easy change as a consequence of experience or of the needs for experimentation. The third and final part discusses issues of human-computer dialogue that must be addressed in the design of a future practical ISIS.

### *Space and Language*

Why do people use maps rather than descriptions of places? An overwhelming number of studies suggest that the human brain deals with information in at least two very different ways (e.g. Bryden, 1982; Moscovitch, 1979; Paivio & Begg, 1981; Taylor & Taylor, 1983). These processes have popularly (and wrongly) been characterized as right-brain and left-brain. There are many other contrasts between the characteristics of the two modes of thought, but for our purposes the best contrast is perhaps "spatial" versus "linguistic."

Space is characterized primarily by the concepts of neighbourliness, ordering, distance and direction. Language is characterized by functional and logical relationships. Space passively exists; language actively moves toward a goal. Space is there, all as a unit; language forms a sequence of cooperating elements. Objects associate with each other by proximity in space; in language they interact. Spatial relations can often be detected in parallel, whereas linguistic ones must be processed one at a time (although several protocol layers may be stacked).

One of the difficulties of specifying just what spatial display does for a user is that the act of analysis tends to destroy the parallelism intrinsic to space. An attempt at a task analysis of map usage prior to designing the first implementation of an ISIS model found mainly tasks that could be performed easily by computer algorithms (McCann, 1982). Most subtasks derived from the analysis were ones that a reasonably competent database system could answer from a simple linguistic query: "How far from A to B;" "How many enemy guns in range of X?" and so forth. After the analysis, few tasks were left for which spatial display seemed the preferred mode, or for which spatial input seemed better than a linguistic query. Intuition says that the task analysis was wrong, and it is precisely for intuition that spatial display is most useful. The analysis did not discover intuition among the tasks the human must perform, because it is not a task; it is an ability.

Spatial arrangements, whether of topography or of military units, do not have any inherent significance. Their significance derives from the association of their parts into meaningful structures. Skilled tacticians, whether they be military tacticians or chess masters, can remember meaningful arrangements of units much more easily than random arrangements. They remember groupings suited for functions such as attack, mutual support, area defence, and so forth, and can reproduce these structures as parts of the overall situation. This kind of pattern recognition is much easier in a pictorial display than in a list (see Fig 1.)[2]

Insert Fig 1 Here

Spatial displays have advantages beyond the presentation of easily recognized structural patterns. Space being continuous, they permit presentation of dense information structures in which neighbouring information is highly correlated. A photographic image of terrain would not be of much use if the height of each point were uncorrelated with the height of its neighbours. In normal terrain, discontinuities (cliffs) occur very seldom, and are noted as structural elements in their own right. A picture of an area in which manoeuvres are anticipated can sometimes be much more informative than a description. The combination of picture, map, and description can together give a much better idea of the possibilities and problems than can any single mode of display. An interactive, dynamic spatial display can

[2] All figures in this paper except Fig. 2 were constructed using the MacPaint software on the Apple Macintosh computer, whose interface techniques inspired some of the examples in the later part of the paper.

provide even greater benefits if animation is used to highlight pictorial information and to show the spatial dynamics of evolving tactical situation (past or projected).

## *ISIS: THE VISION*

Let us imagine a full-scale ISIS based on some future hardware and software technology. What should it do? How should it look to a user? What benefits would it provide as compared to a text-based decision support system?

A commander sometimes works with one or more staff officers, whose specialised expertise and situation knowledge complements his own. He can bounce ideas off these officers, and request detailed information about the current situation. The ISIS should serve as an "electronic staff officer," whose abilities complement but do not supplant those of the commander and his human staff officers.

A prominent aspect of the full ISIS is that its operation should *not* require the user to have any specialized skills. The interposition of specialists between the commander and the ISIS would substantially reduce the value of the ISIS. The problem is the same as that of the helmsman on a ship. The helmsman serves as an intermediary between the Captain's control decisions and the actual manoeuvring of the vessel. In normal conditions, having a helmsman relieves the Captain of routine requirements, but if the Captain has a control console that he can use, he will be able to control the ship precisely under difficult conditions in which control through verbal communication to a helmsman would fail (Lewis, de la Riviere, & Logan, 1966; Lewis & Sweeney, 1960). Similarly, a commander might well use an operator to deal with the ISIS under normal conditions, but should be able to use it directly when under stress or planning an engagement.

In communicating with their staff officers, commanders usually talk and draw annotations on maps. They do not ordinarily type their communications. It is hoped that communications with the ISIS will be in the same modes: talking and drawing. Keyboarding will be held to a minimum, perhaps to be used only by system specialists for control and modification of the ISIS itself.

### *What should a full ISIS do?*

*Decisions and assistants* A decision is a selection among options. The two phases of making a decision are the discovery of options and the evaluation of the options discovered. A decision aid can support either or both. The ISIS is supposed to support both. Evaluation is the easier phase to support, because, having the options, the user knows what questions to ask in order to determine their costs and benefits. An office database management system can support option evaluation; even a spreadsheet can be used to test the financial effects of management decisions. A spreadsheet cannot, however, suggest possible options unless the mere fact of laying out the formulae suggests new possibilities to the user.

The ISIS could support option evaluation by providing algorithms or expert systems to analyze the probable costs and benefits associated with courses of action. At a mundane level, it could assess the terrain areas covered by enemy or friendly artillery or ground-level spotters; it could assess the probable going conditions over the projected area of manoeuvre given the recent weather, and so forth.

The ISIS may be expected to serve its function as suggester of options through its spatial displays, simply because they lay out the situation in a manner unbiased by prescribed relationships. Any linguistic description of a situation is biased by the presumptions inherent in the words chosen: "Attack coming from the North East" presumes that the configuration detected actually represents an attack. The presumption may be true, but it forecloses other possibilities and may tend to bias responses. The response to "attack" is likely to be "defend," whereas the response to a map display of the same situation may be to withdraw to the side, counterattacking the enemy's rear after he has gone past. In this, ISIS does not differ from normal map displays. The spatial nature of the display is the same. ISIS does differ, however, in its ability to give dynamic displays of the history, and in its ability to show many different levels and kinds of information detail on request.

Dynamic displays do for time what spatial displays do for geography and situation description. A situation that evolves over days or weeks might be replayed in seconds, making different levels of movement apparent to the eye. Human perception is much better at detecting coherence in changing patterns than in detecting groupings in static patterns. In some circumstances, a dynamic display might improve on a static one as much as a map improves on a written list of unit positions.

The ISIS talks not only to commanders, but also to information providers and analysts. For this reason, it may be expected to contain more comprehensive information about current situations than would be known to any one officer. It can provide synoptic and dynamic views through its spatial displays, or detailed lists of data through its alphanumeric interface. These displays could be equally useful to the specialist staff officers and to the commander, and could serve as a medium of conversation in the same way as does a briefing map. If the ISIS contained expert subsystems for various purposes, these experts could be called up to answer questions in their realms of expertise — perhaps logistical, perhaps on the physiological and psychological capacities of troops given the recent history of battle stress, perhaps on likely enemy reactions to proposed moves, and so forth.

The obvious uses of the ISIS, as described above, are to lend its information and computational resources to the commander and his staff, so that they can test their ideas, or enter new information, or request the performance of some computation. The mere presentation of spatial displays is sufficient to give ISIS a second function, the development of new ideas in the commander's head. This function of ISIS might be enhanced by the provision of a special kind of expert — the "Stupid Staff Officer" — who has been given the name "Ludwig," for a reason lost in the mists of time.

Ludwig is expert in asking questions relating to the topics that have recently occurred in the interaction between the commander and the ISIS. Ludwig knows very little about strategy, geographic information, or much else in the database (a little, but not much). An early demonstration of false "Natural Language" interaction was the Eliza-Doctor program (Weizenbaum, 1967), which acted like a Rogerian psychotherapist. Eliza could transform statements by the user into the form of questions, and would occasionally introduce elements into the dialogue from earlier in the session:

User: I feel bad today.
Eliza: What makes you feel bad today.
User: I had an argument with my brother.
Elize: Does your brother make you feel bad?
....

Ludwig is a modern incarnation of Eliza. Its natural language understanding capabilities would be greatly enhanced, as would be its ability to incorporate elements from the past dialogue and from other possibly similar dialogues. Its small amount of world knowledge would be based on military concepts, rather than on psychotherapeutic catchwords. Its job would be to ask the commander questions that might occasionally trigger a novel thought, or make him see the situation in a new light, much as a bright graduate student can ask the stupid question that solves the professor's dilemma. Some commanders would never talk to Ludwig, but others might find him quite helpful in releasing from their brains things they might not otherwise connect.

The intention of the full ISIS is that the human will use human intuition, based on the most accurate facts and computations that the computer can provide. ISIS should not substitute for human skills, but should supplement them with abilities that humans do not have, and help humans use the knowledge and skills they do have.

*Some required capabilities* From the user's point of view, ISIS serves many purposes. Even without using the methods of artificial intelligence:

- It should compute the answer to direct questions such as "How far is it from A to B, using routes invisible from C."
- It should serve as a repository of information such as "There exists a class A road named Highway 262 that follows these coordinates ...." or "Smoke has been reported from Hill X."
- It should on request display information about regions, about types of entities, about battle situations.
- From spatial information it contains, it should be able to synthesise or select visual imagery showing what certain places would look like from cer-

tain viewpoints.

- It should compute mobility conditions for selected areas, taking into account recent weather.

All these functions (except possibly for the synthesis of visual imagery) can be performed by a simple relational database mechanism, such as that incorporated in the DCIEM Spatial DataBase Management System 1 (SDBMS 1). Special functions or algorithms are needed for computation of intervisibility or mobility, but such functions can be easily mated with a database management system. Much more becomes possible with the inclusion of various aspects of artificial intelligence, such as a Natural Language Interface (NLI); expertise in logistical requirements, military doctrine, and so forth; and particularly intelligence in the system's approach to dialogue:

- If an item of information creates an apparent inconsistency (e.g. a position for a tactical unit is incompatible with previous information), the system should be able to request confirmation, giving reasons for its request and attempting to discover whether the difficulty is in the new information or in some aspect of the old. When the new information is correctly entered, with explanation, corrections of the inconsistencies should have a ripple effect throughout the affected areas of the data structure, and should cause warnings and other messages to be sent to humans whose previous use of the data might have been adversely affected by false "knowledge."
- The system should keep track of both its interlocutors and their responsibilities, so that it can respond appropriately to implied questions. The mode of response should be appropriate to both the enquirer and the query.
- The system should have some knowledge of the rules of military engagement, so that it can project the likely outcomes of suggested moves. It should be able to show different possibilities as well as the most likely outcome, and should be able to keep in memory any contingency plans developed by the commander as a response to various projected outcomes.
- On a more mundane level, the system should be able to compute the logistic requirements of plans, and determine whether they can be met with the resources at hand within the time scales demanded. It should be able to answer queries about the availability of resources of all kinds, whether they be derived from higher levels or authority (e.g. tactical air strikes) or computable from stored information (e.g. artillery capability for attacking particular enemy formations).

Part of any human-human protocol is timeliness of responses: A slow response indicates some significant activity on the part of the partner. Delayed responses could cause loss of fluency in the development of ideas, both because the human might perceive the delay as meaningful, and because it might interrupt his train of thought. The database management system must therefore be fast, whether it is simply retrieving information stored in an immense database, or performing

complicated projections of enemy responses to planned moves.

### *Some conclusions about ISIS*

*Insecure information* Any mature decision support system in a command and control environment should maintain within itself a description of the current situation, so far as it is known. From this statement flow several consequences:

- the system must be able to acquire information from many sources, whether they be sensor systems, staff officers, or planners;
- information sources may be mutually contradictory, perhaps because the situation changed between the entry of two pieces of information, perhaps because of error, or perhaps because of incompleteness of individual data sources;
- a description of the situation implies some means of relating information from different sources, and therefore some means of detecting and allowing for uncertain, inconsistent, or changing information.

The database must be managed dynamically, in the sense that new information is checked with existing information both for consistency and for timeliness. New information should be consistent not only with existing data on "the same" aspect of the situation, but also with aspects that can be inferred using the new data. Such inference can be very straightforward (e.g. A land unit has been reported at location X,Y, which is known to be in the middle of a lake) or complex (e.g. The enemy unit that has been at X,Y for the last week seems to have moved. This is inconsistent with General YYZ's normal procedures. The enemy has probably had a change of command, and the significance of further moves must be re-evaluated.) Obviously, algorithms can detect some inconsistencies or implications, and artificial intelligence may be used for others, but human intuition and expertise is necessary for the more complex ones. There may, however, be no particular reason for choosing one alternative over another to resolve an inconsistency; both may have credible support.

If there is no strong reason for accepting one from a selection of plausible but mutually contradictory pieces of information, the system should not force a resolution, but should annotate as being possibly unreliable any user interactions that depend on them. It should also examine new incoming information, to assess whether it bears on unresolved conflicts.

Conclusion (1): the system must have internal means of performing consistency checks and catching relevant implications of data. It is an "intelligent system." It must be able to tolerate the continued presence of apparently inconsistent information, removing the inconsistency only when there is adequate evidence to do so with confidence.

*ISIS as a communications medium* The ISIS serves as a communication medium, not only between sources of information and the commander, but also from the commander to the people he controls. It has many input and output positions, and one use that it can make of them is to show many people the same view of a situation. When several of these people each have the ability to enter information, so that others can see what has been done, ISIS provides a "common visual space." Through this common visual space, the commander can describe his plans, request situation estimates, propose options, and so forth.

Part of the "situation" whose description is held in the system consists of the commander's immediate plans and contingency plans. It also includes information about doctrine, which determines actions that are not specifically incorporated in the commander's plans. Doctrine can be considered as default specifications for action in specified circumstances; the commander can override these defaults whenever it seems appropriate to do so.

Conclusion (2): the system's expertise should include as background the ability to analyze situations and determine which aspects of doctrine are applicable, as well as to incorporate the specific plans of the commander, both actual and contingency.

*Spatial thought and spatial display* Humans think in at least two characteristically different modes. These modes go under many different names, which reflect various aspects of their differences in character. No one name adequately defines the differences in the modes, since they very seldom are apparent to introspection, or express themselves in overt behaviour. One facet of the difference might be characterized as linguistic vs. spatial. Another is sequential vs. parallel, although both modes of thought are probably performed as a pipelined sequence of parallel feedback processes. Other contrasts are time-bound vs. timeless; logical vs. associative; rule-based vs. pattern-based; functional vs. content-based. Different people tend to emphasise one mode or the other differently in their thinking.

The spatial mode of thought enables humans to connect different aspects of patterns into a coherent whole, to notice consistencies, to recognize analogies, to "see the big picture." It is the intuitive half of thought. The logical mode, however, permits the construction of sequences, fine discriminations, interrelation of multiple contingencies, and the extraction of correct interpretations from a multitude of possibilities. The logical mode is much better than the spatial in analyzing the details of complex situations. Spatial display helps the human to take advantage of the intuitive mode of thought, whereas language (including the language of mathematics) gives preference to the logical mode.

Spatial display and the display of spatial data are not the same. Almost any information can be displayed spatially: financial data, for example, is often better appreciated as a bar chart than as a list of figures. Spatial data are those whose relationships depend on their location rather than on their function. Spatial data can be geometric or topological. Geometric data have measurable distances and directions between objects; both distance and bearing can be measured on a

continuous scale. The space containing the data is "dense." An object could be placed at any location, though in practice there is a limit to the resolution of measurement, which limits locations to points on a fixed grid. Topological data have no measurable distances or directions. One object may be between two others, or to the left or right of some arc, or inside a closed curve, but it cannot be "further" from A than from B unless B must be passed on the way to A.

The distinction between the types of spatial information is well demonstrated by the two maps of London most often seen by tourists: an Underground map and a street map with the Underground also shown. The Underground map is topological Fig 2.

==========

Insert Fig 2 Here

==========

Distances between stations are not implied, and neither are relative ground directions except in the most gross manner. What the Underground map does show are the sequences and interconnections of stations. The street map, on the other hand, shows distances and directions as accurately as is convenient. On such a map, the pattern of the Underground is confusing. To provide information about the Underground to a tourist, it is better to ignore the actual distances and directions, and show only the topological relationships. For a subway engineer, the geometric map may be much more useful, since he needs to know the distances and locations of much more than just stations.

Conclusion (3): the ISIS should provide both spatial display and language display, preferably in ways that help them to interact. Spatial information may be shown in a variety of spatial ways, each suited to a different task.

*Spatial display of non-spatial data* It is possible (and often useful) to display non-spatial information in a spatial manner. For example, the Computer Corporation of America spatial data management system (Herot, Carling, Friedell & Kramlich, 1980) contains quite ordinary information about ships, with nothing especially spatial in the information except an outline drawing of the ship's silhouette. It displays the information in such a way that the different ships are shown in a spatial relationship, and the information about a ship is displayed in depth. The MIT "Media Room" (Bolt, 1980) room displays all sorts of information laid out spatially so that the user can "move around" in it, plunging deeper into items of interest, or scanning the display space for useful material. Some of the information is spatial, as when the user dives into a map of MIT to focus on a single building, of which an aerial picture may be displayed, and then into a single office in that building, then into a book lying on a desk in that office. Other information, such as a calendar of events, is non-spatial, but it can be found from spatial memory because it is visually located at a particular place in the large display space. The ancient Greeks taught spatial layout as a mnemonic trick in a similar way. Things to be remembered were imagined as sitting in strategic locations in a familiar room. When an item was wanted, the imaginary room was scanned and the thing "picked up" from its remembered location.

The envisioned ISIS is an intelligent, active, distributed data-handling system, that can accept input from humans in spatial or linguistic form, and from a variety of sensing equipment. It can output information on demand either in spatial or in linguistic form, regardless of whether the information itself refers to spatial matters. For example, a database containing topographic information can be queried in a completely linguistic way: "What is the best route from A to B" can be answered by presenting a map, or by listing the roads and their turnings. In different circumstances, either might be preferable.

Conclusion (4): Non-spatial information can often be displayed spatially to good effect. The spatial location of items or classes of items (such as commands, menus, error indications) should be consistent unless there are specific reasons to the contrary. The visual space that contains them should not be uniform, but should be marked with recognizable identification points. Whether information is acquired or displayed spatially is independent of whether the data are themselves inherently spatial.

### *The Vision: Conclusion*

From the users' point of view, the ISIS is both a repository of easily accessed information and a partner in discussion. It can serve as a drawing board or as an encyclopaedia and an atlas. It is a communications mediator, that could request information from other humans when unable to find or deduce the answers to questions, or that could provide a common visual space for several of its users.

The full ISIS will probably never be realized. It requires both hardware and software not yet available. Its concepts rely on rapidly advancing knowledge in several areas of science and technology: database management, artificial intelligence (two rapidly converging areas), cognitive science, distributed computation and interaction design, to name but a few. As the frontiers of research move, so will the concepts of what constitutes an ideal ISIS, so that it will never be possible to actually build what we think we really want. Real implementations of some of the concepts must be built, if only to test out the validity of those concepts. These implementations will use readily available technology in some areas, and will be innovative only where they must be. SDBMS-1 is the first such implementation at DCIEM.

## *SDBMS-1: A DEMONSTRATION TESTBED*

The Spatial DataBase Management System (Version 1) is a small demonstration project to illustrate the value and practicality of managing spatial data in a slightly extended conventional DBMS. [2] It provides information that might aid the decision processes of a brigade commander doing tactical planning. It is intended as

[2] The work on SDBMS-1 has been done by Collins and Moon, Ltd. in close cooperation with the authors, under contract with DCIEM.

a testbed for experiments, and its implementation emphasizes flexibility for future change.

User interaction with SDBMS-1 is through keyboard, voice, and gesture on the input side, and by alphanumeric screen and pictorial display screen on the output. Voice response is not presently incorporated in the design of SDBMS-1, although it would be an easy extension if the interaction protocols seemed to demand it.

The DBMS is a commercially available relational DBMS called MISTRESS (Rhodnius Ltd.) with extensions that allow the use of a "bulk" data type that can be given any arbitrary format, and external operators to implement new operators for specific applications. Bulk data can incorporate any method of describing spatial data, such as strip-trees, quad-trees, polygonal tesselations, and so forth.

Because SDMBS-1 is based around a conventional DBMS, its interactions are based on the formal query language of the DBMS and the actions the DBMS peforms in response to those queries and commands. It is, of course, possible in principle to alter the human-computer interaction by interposing a layer that translates the Mistress queries and responses to some other form. This has been done to some extent on the output side, but on the input side very little has so far been done to shield the user from the complex syntax of Mistress queries.

SDBMS-1 has been designed and implemented using MASCOT technology (JIMCOM, 1983), so that it should be relatively easy to remove modules and insert new ones, or to interpolate transformations and other functions. It is conceivable that SDBMS-1 may grow incrementally into an "intelligent" system, rather like the proverbial old sock that had been darned in so many places that none of the original wool was left. It is more likely, however, that experience with SDBMS-1 will eventually lead to a complete redesign and rebuild of a different system (SDBMS-2).

## *MASCOT technology*

In order to discuss SDBMS-1 in any detail, it is necessary to give a brief description of MASCOT technology (Jackson, 1977; Jackson & Simpson, 1975; Simpson & Jackson, 1979). MASCOT stands for Modular Approach to Software Construction Operation and Test. It was developed at the Royal Signals and Radar Establishment, (UK) in the early 1970's and was established as a standard method of developing concurrent real-time embedded systems for the British military. MASCOT can work as a paper design method, or it can be carried through to the implementation of a running piece of software if two things are provided: (I) a MASCOT kernel that provides six primitive functions, and (II) a language that can compile sequential activities and can call these six primitive functions. A typical MASCOT kernel is only a few hundred words of machine code.

A MASCOT design contains three kinds of entity: Activities, Channels and Pools. Each is designed independently of the others, and can be tested before being connected into the network in which it is supposed to operate. Activities are ordinary sequential programs, having no special properties other than being able to connect to Channels and Pools. Activities never connect directly to each other. Channels pass messages. Once a message has been consumed from a Channel, it is gone and cannot be re-read or seen by a different reader. A Pool, on the other hand, contains information that is stable until changed. It can be read by many different readers, or many times by the same reader. Channels and Pools provide for synchronization across Activities by means of primitives called "Wait" and "Stim." A "Device" might be considered a fourth kind of object, whose job is to connect the MASCOT network to the outer world, but Devices are more easily thought of as Activities with only one input or one output to the net (source or sink).

MASCOT Activities, Channels, and Pools are connected together in an "ACP diagram," which defines all legal means by which data can be transmitted through the network. The MASCOT diagram says nothing about which activities run on what machines, or even about the functions of its entities. The actual programming is kept strictly distinct from the concurrency aspects of the design. All MASCOT activities can in principle run simultaneously, and if some restriction on this concurrency is desired, it must be built into the data flows of the network. If two communicating Activities run on different machines, their communication is by way of a Channel or Pool that spans the inter-machine boundary. The Activities need have no idea that they are on different machines. The hardware dependencies are all built into the Channels and Pools.

A MASCOT diagram shows only what Activities exist, and whether they are connected to one another by a transient Channel, a stable Pool, or not at all. To understand what happens in the structure, one must know the basic functions of the Activities, and the properties of the Channels and Pools, especially as regards synchronization.

## *The MASCOT structure of SDBMS-1*

The MASCOT diagram for SDMBS-1 is shown in Fig 3.

Insert Fig 3 Here

Activities are shown as circles, Channels by I-beams, Pools by stylized cross-sections of a pool, and Devices by rectangles. Input devices are shown on the left, output devices on the right, and operating Activities in the middle. Text-oriented entities are shown in the upper half of the diagram, graphic-oriented entities in the lower half.

The core of the SDBMS-1 MASCOT diagram is the Mistress DBMS and its connected Activity "DBDOER." DBDOER is implemented as a set of routines that access the database directly, rather than through the conceptual Channels that are shown in the diagram. For efficiency, this kind of compression is acceptable in MASCOT design, provided that the data flows among activities remain clear. DBDOER is the interface between the requests from the outer world and the database itself. It interprets the requests to the DBMS, and interprets the resulting data so as to display them appropriately (as a function of display scale, for example).

The main input to the DBDOER is through the PARSER Activity, which accepts three streams of input, from keyboard, voice, and gesture. Gestures may be selections from menus, or drawings that mark traces on the displayed map. In the first case, the PARSER receives text items appropriate to the menu selection. In the second, it receives nothing, but a signal is made ready to indicate on demand that a drawing is available for use. Keyboard and voice both provide text streams to the PARSER.

At the output side, several complex elements affect the interaction protocols. The major activity is the Graphics Manager, whose job it is to ensure that all the right things appear on the high-resolution display screen. For this it must control a cursor, it must highlight selected objects and display menus, and it must show the graphic results of queries in the form of map-like displays. To achieve these objectives, it needs access to several sources of relevant information, which are provided by the Graphics-prompt channel and the various pools that contain display control or display content information.

### *Data structure in SDBMS-1*

In contrast to conventional geographic information systems, the data representation is based on *objects* rather than on *form*. If information is required about an area, the area must be defined as an object either by drawing a closed curve that defines the area, or by reference to an existing object such as a political region. Because of this commonality of representation, it is possible to refer to spatial things and alphanumeric things in the same ways. Each can be accessed by direct command from the keyboard, by building a query through menu selection, or (in future) by selection from the spatial display. Likewise, the result of any query can be displayed alphanumerically or spatially. In the first version of SDBMS, however, the manner of displaying attributes of objects is predetermined by the contents of the Pool "Style Sheets". Geographic information is displayed spatially, for example, and logistic information alphanumerically. The data designers make an arbitrary decision as to how to display each attribute of each entity in the database. As the system is extended, one of the research questions will be how to control the manner of displaying attributes, from the viewpoints of both the user and the software developer. It is a prime area for the future application of the methods of artificial intelligence.

The objects in the database are stored in the form of relational tables. Each table contains all instances of a single object type, such as roads or tanks. Different object types are stored in different tables. Not only objects, but relations among objects are stored in tables. For example, a brigade may include under its command several battalions, an armoured regiment, and so forth. These entities are not considered to be *attributes* of the brigade, but they are related to the brigade through a relational table, "commands." The interrelationship of the tables for unit and command are shown in Fig 4.

Insert Fig 4 Here

(The relationships indicated are for illustration only. The actual tables in the database are more complex.)

The development of a data model suitable for topographic and tactical data is not trivial. Data design for SDBMS-1 is being done using the Entity-Relationship approach. Before any relational tables are built, the objects and the relations among them are sketched out in the form of a network that shows their interconnections. The efficient design of this network has a strong bearing on the later efficiency of the whole system, both in terms of how easy it is to ask questions and how fast the system can answer them. The development of the topographic data model for SDBMS-1 is discussed in Tuori and Moon (1984).

There are two essentially different kinds of information in the database: permanent and transient. Permanent information includes the kind of thing that might be shown on a paper map; transient, things that an officer might pencil in on a transparent map overlay. In the course of operations, even "permanent" information might change. For example, a bridge might be blown up or an airfield destroyed. SDBMS-1 makes no formal distinction between permanent and transient entities, but attempts to keep them in such a way that the transient information is quicker to search and to change. Some queries frequently asked about permanent information may have their answers stored in an easily retrieved form. One means of doing so is to provide a set of "base pictures," which are images of certain geographic areas that can be displayed rapidly on request.

The main spatial display mode is the overlay. Each query to the database provides some information. If the information can and should be displayed spatially, the query results in a single overlay, analogous to a drawing made on a transparent sheet lying over the base picture. Each overlay contains not only the information on the display screens, but also a historical record of the query that was used. If the area of interest is changed by panning or zooming, the query can be run again either automatically or by specific user request, with reference to things within the new geographic boundaries.

Within the base picture objects cannot be selected or otherwise differentiated from the background, but objects in overlays retain their individuality. An object can be selected for further enquiry or as an example of the kind of thing about which a query is to be made. For example, a tank squadron might be selected for an enquiry about its current status, or as a generic example for an

enquiry about tank squadrons in general.

### *Multimodal interaction in SDBMS-1*

SDMBS-1 supports four basic modes of user input, and two modes of output. The four input modes are keyboard, voice, gesturing (drawing) and picking graphic objects or menu items. The two output modes are graphics and alphanumerics. Both modes can be supported on a single output screen, though it is intended that one screen be primarily for alphanumerics and the other for high-resolution graphics.

Two different voice input devices are currently available: a low-cost isolated-word recogniser (Auricle 1), and a more expensive connected speech recogniser (Verbex 3000). The Auricle 1 has a maximum vocabulary of 80 words, the Verbex, 360 words. In each case, a "word" can be a connected phrase rather than a single word. The restricted vocabulary of the Auricle precludes it from serving as the sole input device for the query system. There are more than 80 names of tables and attributes, not to mention control words needed for query construction. The Auricle will be used mainly for experiments on the effectiveness of interaction using isolated words as compared to keyboard or other means of interaction. The Verbex, on the other hand, has the potential for being used as the sole method for input of alphanumeric information, except where novel words are being introduced to the database.

The drawing mode of user input allows both the description of irregular areas and the insertion of geographic descriptions of new entities. For example, the dividing line between the areas of responsibility of two units can be drawn directly "into the database." The graphic description of areas can be used for selection purposes: "Show the enemy guns that can be brought to bear on this area." It can also be used to denote objects already displayed, by describing an area that coincides with the area taken up by the object. In the first implementation of SDBMS-1, the minimum enclosing rectangle (MER) of the gesture and of the object are used for selection purposes, but improved algorithms will permit a better means of selecting specific objects. No object can be selected by a gesture unless it is already displayed. This restriction both speeds the selection process and avoids the possibility that some object hidden in the database matches the gesture better than the object intended. Objects may, however, be selected from the database by means of a query that depends on characteristics of a selected object: "Show me all the members of *this* class."

Multimodal input presents interesting problems both for the implementer of the interface and for the psychologist who tries to understand what the human is actually doing. Four different message streams emanate from the input devices, three of which go through the same Channel to the main body of the network. From the human's viewpoint, these four streams are asynchronous. It does not matter, within limits, at what moment a keyboard token is entered during voice input. If the keyboard token is intended to fit with the voice stream, it is the job of the PARSER activity to find the right place for it. If, on the other hand, the

voice stream provides a complete query with no place for the keyboard token, then the PARSER should assume that the keyboard input either stands on its own or will fit some later input from one of the other streams. If a menu is displayed, the effect should be the same whether the user speaks the desired menu item, selects it from the screen by a gesture, or types a selector code at the keyboard.

The command that reaches the DBDOER is the same, regardless of the manner of input. Nevertheless, some information should be retained about which mode was used for input, because the human may well react differently to a given output mode as a function of the input mode. A voice response to a voice query is often more natural to the human than is a graphic or written response. SDMBS-1 does not incorporate voice response in its initial configuration, but may add it later.

Output modes in SDMBS-1 are rigidly determined by the data structures, and not by the immediate needs of the human user. One major research problem in ISIS will be to determine how best to display particular kinds of information as a function of the state of the dialogue and the display. The corresponding problem for the system designers is how to implement methods of displaying information in the currently required mode. It may be quite simple for the user to specify an output symbolism or format, but it is unlikely that a user untrained in human factors will be much good at specifying effective displays. The research questions will hinge on how to interpret the user's desires in terms of displays that will have the right results.

### *SDBMS-1: Summary*

SDBMS-1 is intended as a testbed for experiments both on the human aspects of spatial data and on the problems of implementing spatial information systems. It is designed in a modular form that permits any of its elements to be replaced by a new design, possibly a whole new subnet. If need be, it could be distributed across several different processors to improve performance or to experiment with the distributed systems now being considered by military authorities. It is hoped that SDBMS-1 might some day provide ideas that are incorporated into operational testbeds for command and control systems, and to that end it is designed not only for experiments, but also for concept demonstration.

SDBMS-1 uses a conventional relational DBMS as its core, somewhat modified and extended to permit arbitrary spatial operators as part of its query language. Both input and display of the data are controlled by interfacing processes which can deal with alphanumeric and pictorial information. There is, however, no "intelligence" built into the system, and all interaction with the data is through the objects in the database rather than through the spatial organization itself. Inconsistency of data is not tolerated, and must be corrected manually if it occurs. There is no provision for automatic consistency checking, much less for recognition of implied inconsistencies that depend on functional relationships among data (SDBMS-1 would never question information that located an armoured headquarters in the middle of a lake).

SDBMS-1 allows input to be in the form of keyboard, voice, graphic selection, and drawing. Output can be pictorial or linguistic. Input and output can be asynchronous on all streams, permitting the user to talk, draw, and watch simultaneously.

## *THE ROAD FROM SDMBS-1 TO ISIS*

The aspirations of a full ISIS system exist in tension with the practicalities of building working systems under limitations of cost and personnel. In developing such a system, there is a great temptation to use the engineering approach instead of considering the user requirements. This results in the selection of methods because they are available, rather than because they are the ones needed for the job. SDBMS-1 is the expression of what can be built using no new technology, but selecting and modifying well established tools. A full ISIS would require a substantial amount of research in both psychology and computer science to be effective, and it might well require the development of new kinds of computer hardware. An intermediate ISIS should be based on reasonably well-understood techniques, though it might no longer use existing tools.

Any version of ISIS acts as a dialogue partner for a human user. From one viewpoint, all the resources of the ISIS exist in order to support the dialogue. Accordingly, a main staging point along the road from SDBMS to ISIS must be an understanding of the nature of interactive dialogues.

### *The structure of interaction*

One way of describing a dialogue is as a set of layered protocols. The idea of a layered protocol can be explained by analogy. In communication between two computers, the machines may be connected by a wire, by radio, or even by hand-carried tapes or disks. If the computers are connected by wire or radio, the timings and voltages of the signals for "one" and "zero" must be agreed between the two machines. If they are connected by hand-carried media, the formats of the media must be compatible. Once the information has been transmitted from one machine to another, the receiving machine can go ahead and use it, regardless of how it was transmitted. The transmission method, with its associated agreements on levels and formats, is the "physical transport layer" of a layered protocol, and anything done with the data to prepare it for transmission or to interpret it after reception is the job of higher layers.

Within a single layer of a protocol, messages seem to pass from one machine to the other and back again, even though the messages actually are passed down to a lower protocol layer in the sending machine and are received from the

lower layers in the receiving machine Fig 5.

Insert Fig 5 Here

Any one layer has no concern with the manner in which its "virtual messages" are actually sent — by wire, radio, disk or tape — provided that they are sent effectively and quickly enough for their purposes. Protocol layers can be analyzed and designed individually, with concern only for their ability to transmit the messages required by the functions of the same layer and the protocols of the next higher layer.

In a Computer-Human Interaction protocol, the analogue of the physical transport layer is provided by the screens, keyboards, microphones and other hardware of the user interface. A higher layer might contain as its protocols the commands that are typed, spoken or selected from menus, plus the error-correction messages generated by the computer. The commands cause functions to be executed; the contents of those commands are the messages passed from the human to the function element of the command layer in the computer. The results of those functions are the messages passed back within the same protocol layer. This pattern of message-passing is illustrated in Fig 6.

Insert Fig 6 Here

Layered protocols in Computer-Human Interaction are discussed in more detail by Taylor (in preparation). The important aspect to note here is that lower-level protocols must be able to transmit the messages required for the proper functioning of higher levels. In particular, this means that spatial display must be considered as part of an entire message loop in relation to the tasks for which the information is required rather than as a single act of information-passing.

***Protocol as syntax*** Syntax describes the formal structure of a language: what kinds of words exist, and how the relations among those words are signalled. "Meanings" have been described by many words: semantics, semiotics, pragmatics, etc. The distinctions among these terms are often diffuse, and one author may contradict another. Here, the term "content" is used as a general term.

In what follows, "syntax" will be identified with the protocol of a particular layer in a communication heirarchy, whereas "content" is the message passed to higher layers (or received from higher layers in an outgoing message). Hence, what is "content" for one layer may be "syntax" for a higher. Herein, perhaps, lies a source of confusion in the linguistic literature.

The Bilateral Cooperative (BLC) model for human understanding of symbolic information (Taylor, 1981; Taylor and Taylor, 1983) asserts that there are two separate but cooperating "tracks" of processes. One detects the existence of patterns that correspond in some respects to patterns seen previously (known words on a page, known behaviours of a friend); the other uses rules and logic to discriminate

among possible identifications, or to construct useful representations of patterns not previously known. The two tracks are assumed to work at a sequence of interconnecting levels, the rules and the patterns both becoming more complex at the higher levels.

At the level of literal sentence meaning, the pattern track associates the various words with a host of attributes, correspondences, categories, semantic fields (all different words for similar concepts), and through a convergence of association discovers the meaning that holds the words together. Hirst (1983) uses a similar concept in the idea of "Polaroid Words," which attempt to share associations with other words in a discourse, in order to disambiguate their possible meanings.

At the same level of sentence meaning, there are many words whose associations are few and diffuse. These are often called "function words," because their job is different from that of the "content words." They designate (or confirm) the function of each content word in respect to the other content words. They are the protocol elements of the message at that level. Above the level of literal sentence understanding, the protocol elements are no longer part of the message, and it is a well-known phenomenon that syntactically correct recall of a sentence is much harder than recall of the sense.

Sentence meaning is often (usually) the highest protocol level studied in human-computer dialogue. It occurs in formal human-computer protocols in the guise of command interpreters, and in AI as "Natural Language Understanding." Beyond that level lies another domain of current AI research, discourse understanding and dialogue. It is the discourse level that holds the promise of providing intelligent decision aids for the person needing the aid rather than by trained operators. Whether the lower-level protocols are formal languages or restricted versions of natural languages really does not matter. Many different protocols can be substituted for one another underneath any given higher level protocol.

Linguistic interaction is so strongly bound with function protocol that language is often analyzed as if it consisted only of syntax. The content of the messages is ignored, and only the protocols described. In part, this is perfectly legitimate, since the content is the message passed to higher level protocols, and is independent of the protocol elements at the level being analyzed. But in part it is unfortunate, because it leads sometimes to the view that language *is* syntax, whereas much of even the protocol function is performed through the association patterns of the content elements.

Spatial information is biased in exactly the opposite way. Space has few intrinsic functional relationships: inside, behind, near, and so forth. These relationships do not *do* anything. They just *are*. The functional aspects of spatial relationships become manifest only at higher levels, such as a particular configuration of units that denotes an attack formation. Such a configuration becomes memorable when labelled, but it also becomes functional in determining what is happening in the scene displayed. The spatial display can be translated into a linguistic message "Attack coming from the North-East," but not into a useful linguistic message containing the positions of individual units. The useful information comes from the

coherence of the spatial structure as seen by a trained observer. Once it has been understood, it can be passed as a linguistic message to the next higher protocol level, forgetting the spatial protocol through which the message was passed.

The definition of specific protocols for interaction with spatial information is an interesting research problem.

*Layered protocols and multimodal interaction* One important feature of a layered protocol is that each layer can be placed on top of a variety of differently designed and executed lower layers. Furthermore, one lower protocol can serve more than one higher layer at the same time. There may be several parallel protocols at a low level that could equally well pass the same message to a higher level. For example, I could speak, write, or perhaps pantomime a simple message to you. Perhaps the same message can be transcribed in a sequence of pictures, or in several languages that we both know. More importantly, the message can be sent through a mixed protocol: some pictures, some text. It is this kind of mixed protocol that we consider in a multimodal interaction.

For communication with an ISIS, convenient modes include typing (if the user has learned to type), speaking or gesturing (drawing or pointing). The ISIS can communicate with the user by written or spoken text, or through sketches or detailed pictures. In either direction more than one mode can be used at any one time. Spoken commands are very easily mixed with gestures:

"Show me more about THAT town" (pointing to the screen).

In such an interaction, the pointing may come before, after, or at the same time as the word THAT. Synchrony, or even order, does not apply to the protocol of a mixed-mode interaction (for we can consider that the low-level protocol has been divided among the modes).

Some message contents can be passed by a gesture that would be very difficult to pass through a linguistic protocol:

"Show me the guns inside THIS perimeter" (drawing an irregular polygon).

However, the rest of the command is difficult to construct through gesturing, except by the expedient of a selection menu. Menus may often be useful, but they form a yet different, even lower level protocol, whose function is to assist the computer and the user to agree on a possible action.

In a multimodal interaction, any desired effect can usually be obtained in more than one way. The higher layers that produce the effect must interface with the lower layers in a uniform way, insofar as the message-passing limitations of particular lower level protocols permit.

***Functional elements: scoping and modification*** A function is an action that affects or relates one or more objects. For it to be performed correctly, several things must be known. These include what object are involved, what aspects of those objects are to be affected, and how the function is to affect them. With a formal command syntax, all these things are explicitly specified, but problems may arise when higher protocol levels are involved. The selection of objects may need to be restricted (e.g. "Tanks in area B" rather than "Tanks"), the attributes affected by the function may require selection (e.g. "Display track-loading of tanks" rather than "Display tanks"), and the way the function affects the attributes may need to be modified (e.g. "Display trackloading as bar-charts" rather than "Display track loading").

In different terms, some questions must be resolved before the function can do what the user wants: "What are we talking about?" (***object scoping***, or more simply ***scoping***), "What aspects of them are we dealing with?" (***relational scoping*** or ***scoping functional relationships***), and "What are we doing with them?" (***function modification***). By way of a very rough analogy, scoping corresponds to the use of adjectives, relational scoping to the use of prepositions, and modification to the use of adverbs in English sentences.

Each function has its set pattern of objects on which it can act. This pattern is sometimes called a "frame." A frame for a generalized three-relationship function is shown in Fig 7.

==========

Insert Fig 7 Here

==========

Each specific function has its own special characteristics that define what sorts of things can fill each place in the frame, and what fills them by default. How the slots are filled determines the user interface. For example, a simple memoryless command-driven system may require the user to type a specific command such as "Select coordinates from highway where type=AA;" (from the MS query language of the Mistress database management system). In this command, "select" is the function, "highway" the object, anything in the "where" phrase is a scoping restriction on the object, and "coordinates" scopes the functional relationships. No modifier appears in this example, and indeed the MS query language makes little provision for modifiers (except that selected data may be either displayed or entered into a prescribed data table). A different interface would result from requiring the slots to be filled by menu selection, even though the same query might be passed to Mistress in the end.

The means of handling scoping and modification are at the heart of the differences among user interfaces. If the system can determine from previous interactions likely values for scope and modification (and perhaps for function and objects), then it is said to be an "intelligent" interface. If all aspects must be explicitly incorporated in each message, it is a "dumb" interface. Clever design of dumb interfaces can make them easy to use; this mainly involves the provision of standard defaults. For example, the objects of a "show" command are most commonly those whose display would be within the current display window in the absence of a

contradictory scoping indicator. Default modifiers may be stored for showing particular attributes: coordinates would be displayed spatially, and load capacities alphanumerically; names would be shown on a map if coordinates were also being displayed, on an alphanumeric display if load capacities were being displayed, or on both if coordinates and load capacities were both being displayed. This is the approach taken in SDBMS-1.

One way of describing an action is by entering the complete command. Another is by "form-filling," which can be accomplished in a variety of ways, all functionally identical but appearing different to the user. In a form-filling approach, the system shows in some manner the frame whose completion defines the desired action. One method of doing this is a sequence of menus, wherein the first requests, say, the function to be performed, the second the objects, the third selects scopes for the objects, the fourth selects scopes for the functional relationships, and the fifth modifications on the functional relationships. Repeated use of the sub-menus completes the construction of the message.

Another approach to form-filling is to provide some kind of skeleton message in which the human user changes the aspects that are unsatisfactory or incomplete. The initial contents of the message are derived entirely from default assumptions, and those defaults change as appropriate to the parts of the message filled in. An example sequence of events in filling in such a message is shown in Fig 8.

==========

Insert Fig 8 Here

==========

Scoping is one of the more difficult aspects of any dialogue. The question is "Just what exactly are we talking about." The answer may be by definition, as in the examples given above, it may be by analogy or example, or it may derive from earlier elements of the dialogue. In scoping by example, the user selects (somehow) an entity that is in some way "like" what is wanted. The attributes of that item are then used in part to define "like," and the extended attributes used to scope the object of the function. Fig 9 shows this method of constructing the same query as that of the previous example.

==========

Insert Fig 9 Here

==========

In Fig 9, the user wants to select for highlighting all class AA highways in the display window. The system uses "in the display window" as a default scoping restriction. First the user selects a particular highway, using a picking gesture adequate to make the selection. The system displays that section of highway, identified to assure the user that the desired item has been selected. Since the user has not defined a function, a menu of permitted functions (larger than shown in the example) is provided, and the user chooses "select." The system immediately presents a sub-menu from which the user chooses "by attr," meaning that the selection will be scoped according to the attributes of highways. The system response is to ask how the attributes are to be chosen: the same as those of the example highway, by direct entry of desired values, or perhaps by changing the attributes of the selected

highway. The user chooses to select attributes by value, and selects all class AA highways, which are then shown highlighted on the original map. The selected highways can then be used as a object in a further function, such as one that might locate all river bridges crossed by class AA highways.

The view on interaction as frame-filling for the activation of individual functions may be useful, but unless functions can themselves be objects, it is a limited view. Functions should be able to act as objects for other functions, in the way that phrases or even clauses can take the place of nouns in English. Indeed, the example in Fig 9 implies as much, because the Class AA highways selected by the function are left highlighted as potential objects for a new function.

*Spatial data representation*

SDBMS-1 treats space as an attribute of objects. All access to data is through the objects themselves, and only through accessing all objects in a given area can the contents of the area be determined. Objects can be indexed in many ways for rapid access. Location and extent are among these attributes, but space is not natural to SDBMS-1, as it must be in a fully functioning ISIS. In order to answer questions such as "What area can be seen from anywhere along THIS road," SDBMS must extract from the database a digital terrain model covering the whole area that might be relevant. The work is then done on this terrain model. A purely spatial database, however, would be able to access information using location as a primary index. Its canonical question would be "What is here?" rather than "Where is this thing?" The full ISIS would be based on both kinds of canonical question. At present, we do not know how to design a database on such a double foundation. Some of the geographic issues involved in developing an ISIS have been discussed by Moon, Collins and Blair (1984).

*Information zoom* Information zoom is an aspect of the ISIS that has been slighted in the design of SDBMS-1. Information zoom allows the user to access more and more detailed information about things of particular interest, just as spatial zoom allows him to look more closely at an area of interest. In both cases, information of less interest may be lost to the display because of its limited resolution and size. A civilian example of information zoom might go as follows:

- The user requests a particular area to be displayed. Default scoping and modification rules are supplied by the system, and a map is displayed showing water and land areas, plus major transportation routes and urban areas.
- The user requests information on power utilities. Lines are drawn on the map to show the major transmission corridors.
- More detail is requested on the power utilities. Roads are removed from the map, and the symbolism of the transmission corridors changed to show their

voltage and peak load capacities.

- More detail is requested. The symbolism is again changed to note the number of cables and the number of separate tower lines in each corridor. Graphs are added to show normal daily variations in power demand. Probably at this point, only major rivers and cities remain of the original map, and these are probably displayed at low contrast so as not to interfere with the important detail of the power system. Power generating plants have taken the place of cities as focal points on the map, and there are probably special symbols for distribution sub-stations. -

In this example of information zoom, the map underwent no scale changes or alterations in the area displayed. In other examples, both scale and type of information might change. During military operations, it may often be desirable to see the shape of the overall situation, and then to look more closely at the proceedings in critical areas.

*Browsing and information zoom* One of the problems of using a large database is knowing what information might be available. Leafing through a paper Encyclopaedia shows one a large variety of topic headings, some of which could be useful. In a computer database, browsing is at once easier and more difficult than in a paper book. It is easier because indexes can be made available for rapid scanning, and more difficult because rapid movement from one area to another is hard to control.

Browsing and information zoom are intimately related. A browser wants to find out where there might be information that could be related to a current question, or perhaps to find information that might stimulate a useful question. Once a hint of useful information has been discovered, the browser may well want to find out more about it.

The evolving technology of video discs may aid browsing in a complex visual space. The MIT "Media Room" (Bolt, 1980; Lippman,1980) uses video discs not only to show images of locations, but also to show movies of selected scenes. The technology allows selection of arbitrary frames, and it is easy to imagine a more or less random browse through a catalogue of pictures relating to data contained in sections of the database. Pictures might be displayed at a rate of 3 or 4 per second, slow enough for a user to see what they are about, but fast enough to do a lot of browsing in a short time. When an interesting picture appeared, the user could stop the browse and go back to the one that was interesting. Such a browse could be constrained in many ways, by topic area, by geographic region, or by any attribute known to the database.

The structure of information is something that is hard to describe in a paper book. In a computer database, the structure is implicit and can be displayed graphically. The Entity-Relationship (ER) diagram is used in designing data models for SDBMS-1. It provides an overview of what kinds of information exist in the database, and how they are related. An example is shown in Fig 10.

Insert Fig 10 Here

The ER diagram shows only the different kinds of objects about which information is available. A data dictionary gives information about the structure of each object, such as what attributes are mentioned and what kinds of data types these attributes might be (alphanumeric, integer, coordinates, image, and so forth).

Imagine an interaction using a graphic depiction of the ER diagram as a basis for scanning the database. In this example, italics indicate user intentions.

1. *Request display of data structure*: This can be done by any of the methods mentioned: keyboard or voice command, menu selection, or perhaps by graphical interaction. The result is a display of the top-level ER diagram.
2. *Look at "Unit" table to see what information it has*: The user points at the Unit symbol of the ER diagram. The symbol is overlaid by a list of the attributes in the table and their types. ( Fig 11).

Insert Fig 11 Here

3. *Look at the commonalities between the Unit table and the Command relation*: The user points to the line joining the symbols, and it is overlaid by a list of those attributes that they share, and which make sense as the basis of a query. ( Fig 12).

Insert Fig 12 Here

4. *Find out what values occur in the data for a given attribute*: The user points to the attribute in the list displayed. The list is overlaid with another list of all the values that occur. ( Fig 13).

Insert Fig 13 Here

5. *Find values for particular data entries (e.g. How much ammunition for 105mm guns does the 23rd Artillery Regiment have?)*: The sequence of operations here could be:

   - Select "Uses" and "Owns" relations that connect "Ammunition"

with "Weapon" and "Unit" ( Fig 14).

Insert Fig 14 Here

Each table displays the available attributes, highlighting the name of the entity tables it connects. Notice that "Ammunition" is shown in the "Owns" table, although other things can be owned. The selected line connected "Owns" with "Ammunition," so that it can be unambiguously shown as the owned object type. Also notice that although it was not selected, the table "Ammunition" is displayed with "Type" highlightes, because it is a common link between the two relations that were selected.

- From "Weapon.type" select "105mm how", and under "Unit.type" and "Unit.ID" enter "Arty.Reg" and "23". This could be done from keyboard or by selection from a displayed list. Also select "Holdings" from the "Owns" table, to show that this is the information desired. The query is now specified, as shown in Fig 15.

Insert Fig 15 Here

- The answer is displayed as a table showing the elements that led to the query, and the data satisfying the query. The entities in the query were directly fixed by the user; the answer consists of the items selected but not fixed. ( Fig 16).

Insert Fig 16 Here

6. *Find information about an enemy unit identified only by location*: In the previous example, the actual identification of the units was known. This may often not be the case for an enemy unit; the commander may need to point to the unit on a map, or refer to it by some characteristic. In such a case, the interaction sequence may be similar to that of the previous example until the final selection. At that time, the user might call up the map (which he had presumably been looking at before starting the query) and simply point at the unit in question. The pointing gesture would select the unit in the same way that pointing at its representation in a table would do. ( Fig 17).

Insert Fig 17 Here

*Active Objects* One of the underlying presumptions of the discussion has been that any object visible on the screen could be selected. Selection of an object might cause an action to happen, or might simply make that object available for use in some other action. This applies to elements of a menu, to symbols representing a relation, or to things on a map display. What happens when a thing is selected must depend on the state of the human-computer dialogue, and on the mode of selection.

The biggest problem with the concept of active objects is that of scoping. How does the computer system know just which of several overlapping objects is intended when a gesture attempts to designate one? Scoping, here, is considered at a low protocol level, because the action that occurs when an object is selected may well be a component of a higher-level message. This fact provides a clue as to how to determine the object intended by a gesture. The current state of the higher level message may restrict the possibilities, or if the message is being formed asynchronously through another low-level medium (e.g. voice), the correct selection may become evident after a little while. If ambiguity persists, a high-level error-correction protocol may be invoked to clarify the issue.

As with all protocols, it is a good idea to reduce the number of errors as low as convenient, but not a good idea to go to great lengths to eliminate errors. The selection protocol should therefore be designed so that ambiguities seldom occur in the original gesture, but when they do, they should be disambiguated through interaction at a higher level.

## *An intermediate ISIS*

A full ISIS is beyond our reach. SDBMS-1 is largely designed and in part implemented. What should be the design goals for an intermediate ISIS (SDBMS-2)?

SDBMS-2 should above all be interactive. This goal implies that work should concentrate on the multimodal dialogue protocols, and on speed of response from the database. It should incorporate some level of intelligence in its dialogue protocols, so that higher protocol layers can be implemented than are possible using algorithmic methods. It should be a good system within which to browse, because browsing often provides unexpected options and shows novel patterns in the data. It should provide a wide range of spatial operations, especially in the realm of quasi-real imagery of terrain. It should be able to show real imagery of selected views, if it exists, and to transform real imagery to new viewpoints when appropriate. It should be able to synthesize views from topographic data where real images do not exist. Such capabilities are routinely available in several current systems, although the synthetic imagery sometimes looks a bit like a cartoon.

SDBMS-2 should be able to make "intelligent" interpretations of some facets of its data. For example, if the recent weather has been wet, it should be able to deduce that low-lying areas may be muddy or marshy, and to include that information when the relevant areas appear in the answer to a query.

SDBMS-2 may not be an intelligent partner to the human user, but it should provide a testbed for experiments on how to incorporate intelligence in specific areas. It should include specialized expert subsystems that serve either to control its own operations or to work with the stored data. It should be able to perform some limited consistency checking and cleaning of new data, preferably rapidly enough to allow false data to be highlighted at the time it is entered. SDBMS-2 should bring some intelligence to levels of dialogue protocol above the direct command.

SDBMS-1 does not provide "triggers:" actions that are performed automatically when certain states or operations occur. For example, it cannot change the number of tanks available to a brigade when tanks belonging to a subordinate unit are disabled. The fact that the number has changed is discovered only when an explicit query causes the calculation to be made. A trigger could generate a warning when the total number available fell below some threshold. SDBMS-2 should implement triggers based on arbitrary functions of the data, including functions that involve some "intelligence" to perform.

The goals of SDBMS-2 could probably be achieved largely with existing technology, but not with off-the-shelf components. The research areas needing most attention appear to be:

1. The nature of human interaction with spatial as opposed to linguistic information.
2. Speed of response, coupled with intelligence of response.
3. The application of artificial intelligence and expert systems.
4. Multi-level, multi-modal dialogue structures and protocols.
5. The analysis of data consistency and the provision of warnings when queries depend on inconsistent data.

Other areas will doubtless become apparent as the work progresses.

## *CONCLUSIONS*

Spatial information systems provide novel opportunities for decision aiding. Not only can they assist in the evaluation of options, but also they can help the discovery of options in a way that linguistic information systems cannot. Humans manipulate information both associatively (spatially) and logically (linguistically). These two modes are complementary, and together provide a power that neither alone can achieve.

Modes of display do not intrinsically match modes of thought. Spatial information can be analyzed logically, and verbal information can be built into a "big picture." Nevertheless, spatial information often demands spatial manipulation in order for the human to "see what is happening." The computer, on the other hand, is best at logical manipulation. Spatial information systems should show their power in the symbiosis of computer logic with human intuition.

Dialogues can be described in terms of a layered protocol, whether they be between humans, between computers, or between a human and a computer. The computer-human interface consists of the specific protocols used by one or more layers. Classical human engineering of workstations can provide effective low-level protocols, but their effectiveness cannot really be judged unless higher protocols are defined. The usefulness of a protocol is determined by the ease with which the virtual messages of higher levels can be transmitted. Cognitive engineers must develop the higher protocol levels. Whether the desired protocols can actually be implemented in the computer depends on the development of AI techniques. A full interactive spatial information system probably depends on ideas yet to be discovered.

A low-level prototype Spatial DataBase Management System (SDBMS-1) has been largely designed and implemented in part. It will serve as a testbed for ideas on the computer-human interface, and for studies of the presentation of spatial information. It is based on an extended relational database management system, and uses the MASCOT technology for its implementation. SDBMS-1 should be easy to modify and extend as the need arises.

An intermediate ISIS should incorporate some intelligence, both in its control of dialogue, and in its interpretation of the data it contains. Although it is not expected to be an "electronic staff officer," in the way a full ISIS would be, it should be able to provide some inferential support to its users and to provide warnings about inconsistent or unlikely data.

## *FIGURE CAPTIONS*

**Figure 1. Similar unit information displayed on a map and as a list of coordinate locations.**

**Figure 2. A topological map of the London Underground system, showing the order and linkage of stations, but no diostanceistances or bearings.**

**Figure 3. The MASCOT diagram for SDBMS-1.**

**Figure 4. The relationships of a brigade and two battalions in the "Units" and "Command" tables.**

**Figure 5. Layered protocols. Messages seem to pass from one machine to the other at a single protocol level at a time, but in fact all messages pass down from the originating sender to the bottom level and thence up the receiving chain to the receiver at the same level as the originating sender.**

**Figure 6. One layer of a layered protocol. Protocol elements are circulated from one sender to the complementary receiver, from there to the corresponding sender and back to the originator's sender. Messages causing functions to be executed are passed from the function area to the corresponding function area on the other side, through the protocol handlers.**

**Figure 7. The frame for a function with three relations. Each object can be scoped, the function can be modified, and the relation between the function and itany object can also be scoped.**

**Figure 8. Filling in a default message.**

**Figure 9. Constructing a query by means of a variety of spatial interactions. In the final stage, the keyboard is invoked to choose a particular type of highway for highlighting.**

**Figure 10. A small part of an Entity-Relationship diagram. Entities are shown by ovals, relationships by parallel bars.**

**Figure 11. Finding what is in the "Unit" table.**

**Figure 12. How does the "Unit" table relate to the "Command" table?**

**Figure 13. Finding the values available for a given attribute.**

**Figure 14. Beginning a complex query. Step 1: select the relationships likely to connect the entities of interest. Intermediate entities are automatically selected and the link attribute(s) highlighted.**

**Figure 15. Continuing the complex query. Desired values are selected for scoping the query.**

**Figure 16. The answer is displayed.**

**Figure 17. An alternate way of scoping one of the elements of the query, by selection from a spmap display instead of the schematic. This method is often appropriate, but especially when the identity of the desired element is unknown.**

## *REFERENCES*


Bolt, R.A. "Put-That-There": Voice and gesture at the graphics interface. *Computer Graphics*, 1980, *14*, 262-270.

Bryden, M. P. *Laterality: Functional asymmetry in the intact brain.* New York: Academic Press, 1982.

Herot, C.F., Carling, R., Friedell, M., & Kramlich, D. A Prototype Spatial Data Management System. *Computer Graphics*, 1980, *14*, 63-70.

Jackson, K. Language design for modular software construction. In B. Gilchrist (Ed.), *Information processing 77.* Amsterdam: IFIP, North-Holland, 1977.

Jackson, K., & Simpson, H. R. *MASCOT - A modular approach to software construction operation and test.* Technical Note 778, Royal Radar Establishment, Oct., 1975.

Lewis, R. E. F., de la Riviere, W. D., & Logan, O. Sea trials of direct bridge control of engines and helm. In *Proceedings of the Ship Control Symposium* (Vol. 1). Annapolis, MD: USN Marine Engineering Laboratory, 1966.

Lewis, R. E. F., & Sweeney, D. M. *Direct bridge control of engines and helm on a destroyer: Sea trials results.* Defence Research Establishment Toronto Report 718, 1969.

Lippman, Andrew Movie-Maps: An Application of the Optical Videodisc to Computer Graphics. *Computer Graphics*, 1980, *14*, 32-42.

McCann, C. *Classification of map-reading tasks.* DCIEM Report 82-R-59, DCIEM, 1982.

Moon, G. C., Collins, S. H., & Blair, G. *Considerations for the design of an interactive spatial information system (ISIS).* Contract Report 8SE80-00052, DCIEM, 1984.

Moscovitch, M. Information processing and the cerebral hemispheres. In M. S. Gazzaniga (Ed.), *Handbook of behavioral neurobiology* (Vol. 2). New York: Plenum, 1979.

Paivio, A., & Begg, I. *Psychology of language.* Englewood Cliffs, NJ: Prentice-Hall, 1981.

Simpson, H. R., & Jackson, K. Process synchronization in MASCOT. *The Computer Journal*, 1979, *22*, 332-345.

Taylor, I., & Taylor, M. M. *Psychology of reading.* New York: Academic Press, 1983.

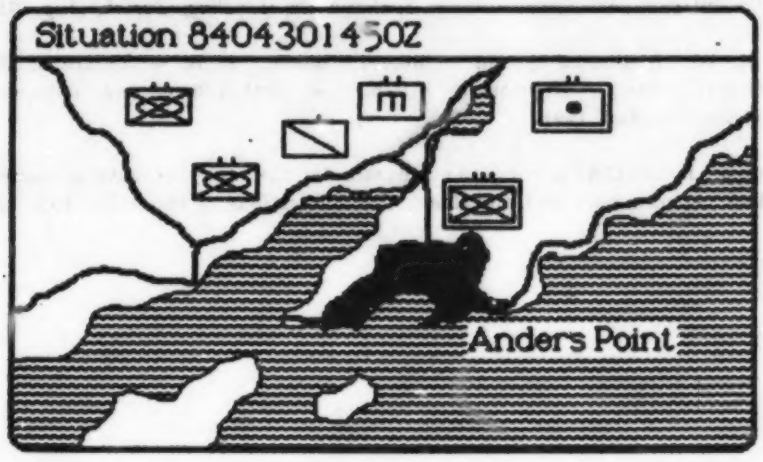
Taylor, M. M. The Bilateral Cooperative model of reading: A human paradigm for artificial intelligence. In A. Elithorn, & R. Banerji (Eds.), *Artificial and human intelligence.* Amsterdam: Elsevier North-Holland, 1984. Also available as DCIEM Research Paper 81-P-4, DCIEM, 1981.

Tuori, M. I., & Moon, G. A topographic map conceptual data model. In *Proceedings of the International Symposium on Spatial Data Handling, Zurich, Aug 1984.* : , 1984.

Watson, R. W. Distibuted system architecture model. In B. W. Lampson, M. Paul, & H. J. Siegert (Eds.), *Distributed systems — architecture and implementation.* Berlin: Springer-Verlag, 1981.

Weizenbaum, J. ELIZA: a computer program for the study of natural language communication between man and machine. *Communications of the ACM,* 1967, *9,* 36–45.

## Spatial Format




## List Format

Situation 84043014502

Friendly

| | |
|---|---|
| 5 Mech. Inf Bn | TE860437 |
| 7 Arm.Inf Bn | TE882417 |
| 6 Recce. Sq. | TE905430 |
| 2 Eng. Bn. | TE926440 |

Enemy

| | |
|---|---|
| Arty. Reg | UE303438 |
| Arm.Inf Reg | UE068402 |

*Figure 1*

*Figure 2*

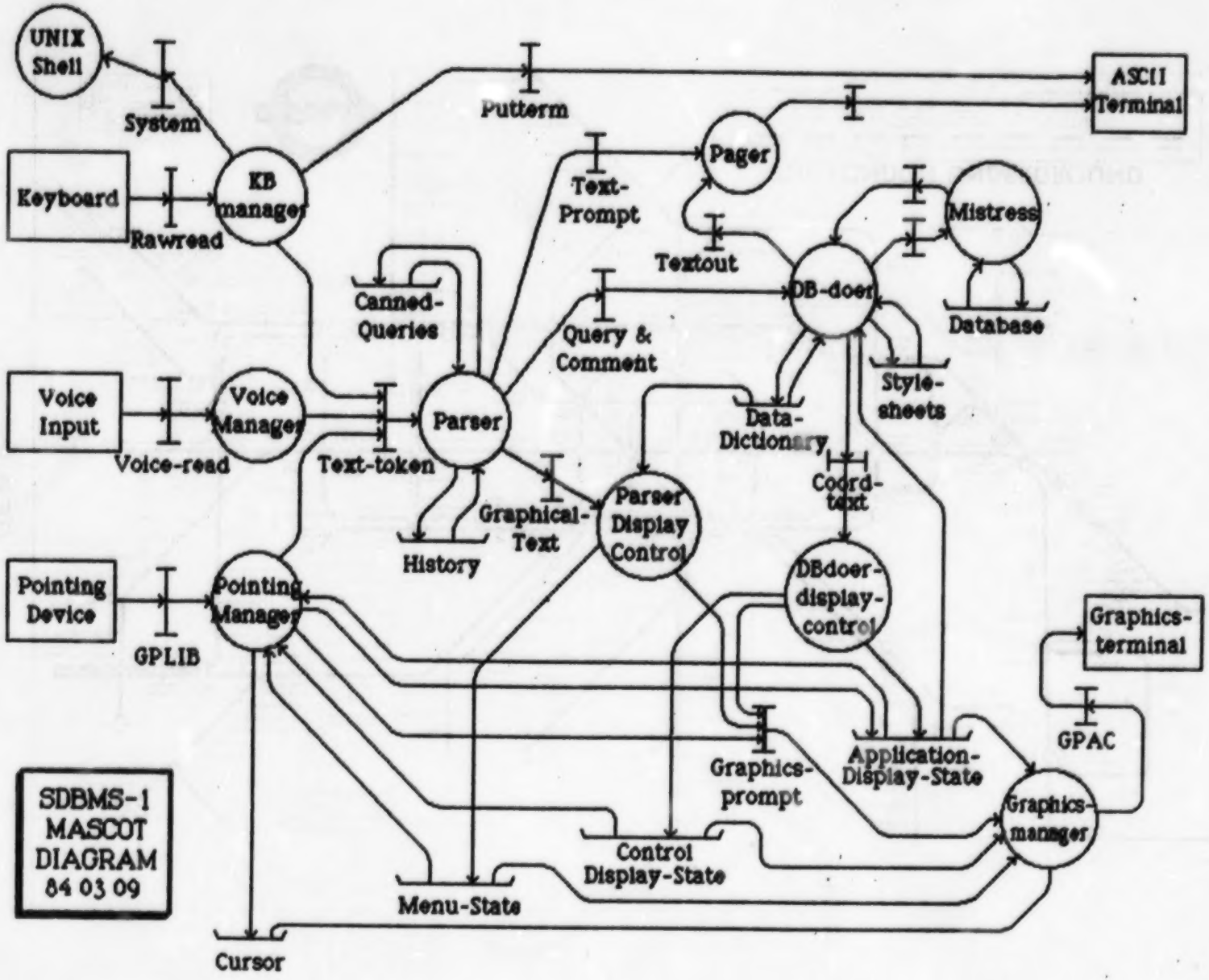
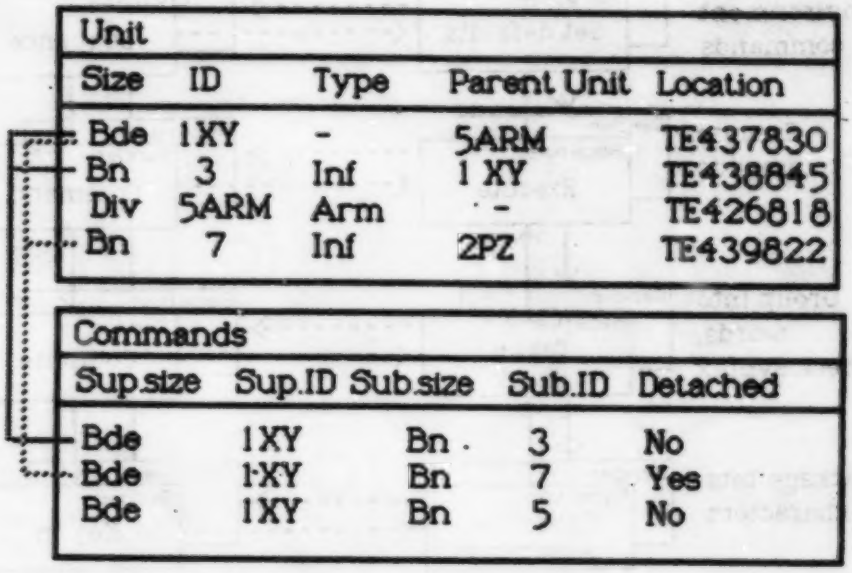

UNIX Shell
System
Keyboard
Rawread
KB manager
Putterm
ASCII Terminal
Pager
Text-Prompt
Textout
Mistress
Database
DB-doer
Canned-Queries
Query & Comment
Style-sheets
Data-Dictionary
Coord-text
Voice Input
Voice-read
Voice Manager
Text-token
Parser
Graphical-Text
History
Parser Display Control
DBdoer-display-control
Pointing Device
GPLIB
Pointing Manager
Graphics-terminal
GPAC
Graphics-prompt
Application-Display-State
Graphics-manager
Control Display-State
Menu-State
Cursor
SDBMS-1
MASCOT
DIAGRAM
84 03 09

Figure 3



Unit

| Size | ID | Type | Parent Unit | Location |
|---|---|---|---|---|
| Bde | 1XY | - | 5ARM | TE437830 |
| Bn | 3 | Inf | 1 XY | TE438845 |
| Div | 5ARM | Arm | - | TE426818 |
| Bn | 7 | Inf | 2PZ | TE439822 |

Commands

| Sup.size | Sup.ID | Sub.size | Sub.ID | Detached |
|---|---|---|---|---|
| Bde | 1XY | Bn | 3 | No |
| Bde | 1XY | Bn | 7 | Yes |
| Bde | 1XY | Bn | 5 | No |

*Figure 4*

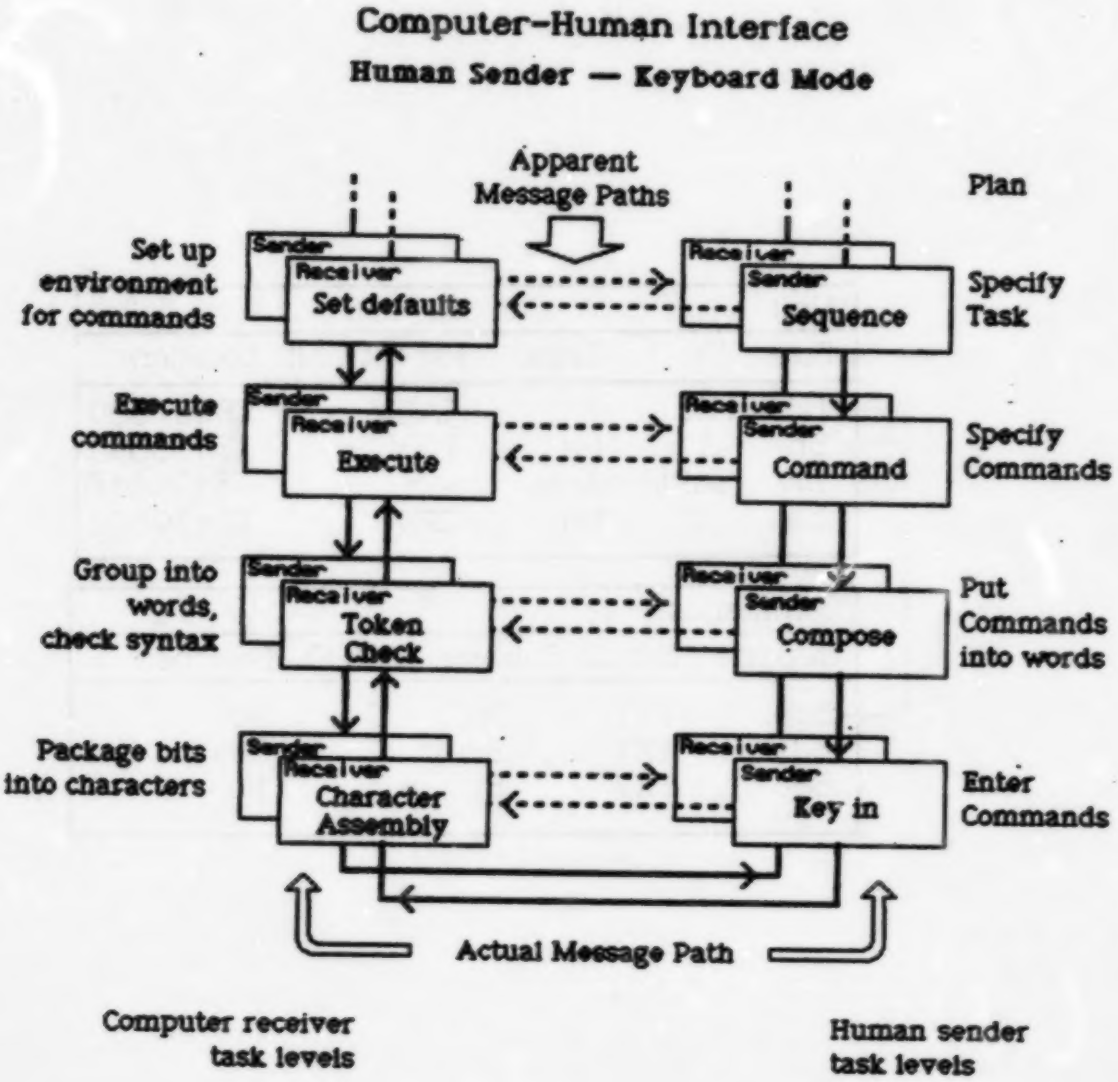

Computer–Human Interface
Human Sender — Keyboard Mode
Apparent
Message Paths
Plan
Set up
environment
for commands
Sender
Receiver
Set defaults
Receiver
Sender
Sequence
Specify
Task
Execute
commands
Execute
Command
Specify
Commands
Group into
words,
check syntax
Token
Check
Compose
Put
Commands
into words
Package bits
into characters
Character
Assembly
Key in
Enter
Commands
Actual Message Path
Computer receiver
task levels
Human sender
task levels

*Figure 5*

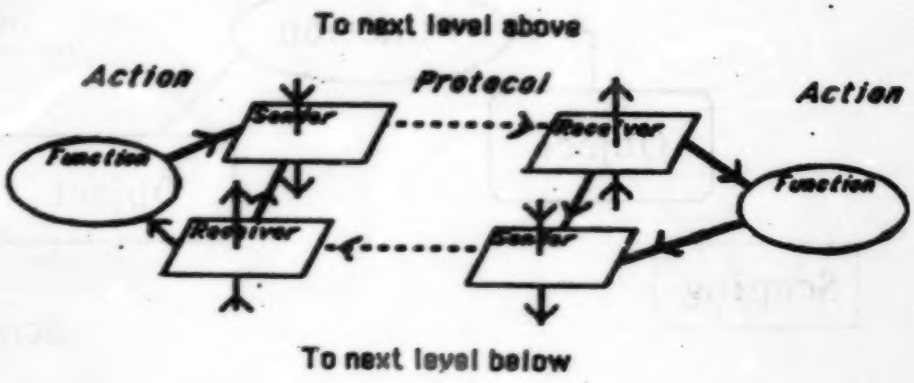

To next level above
Action
Protocol
Action
Function
Sender
Receiver
Receiver
Sender
Function
To next level below

*Figure 6*

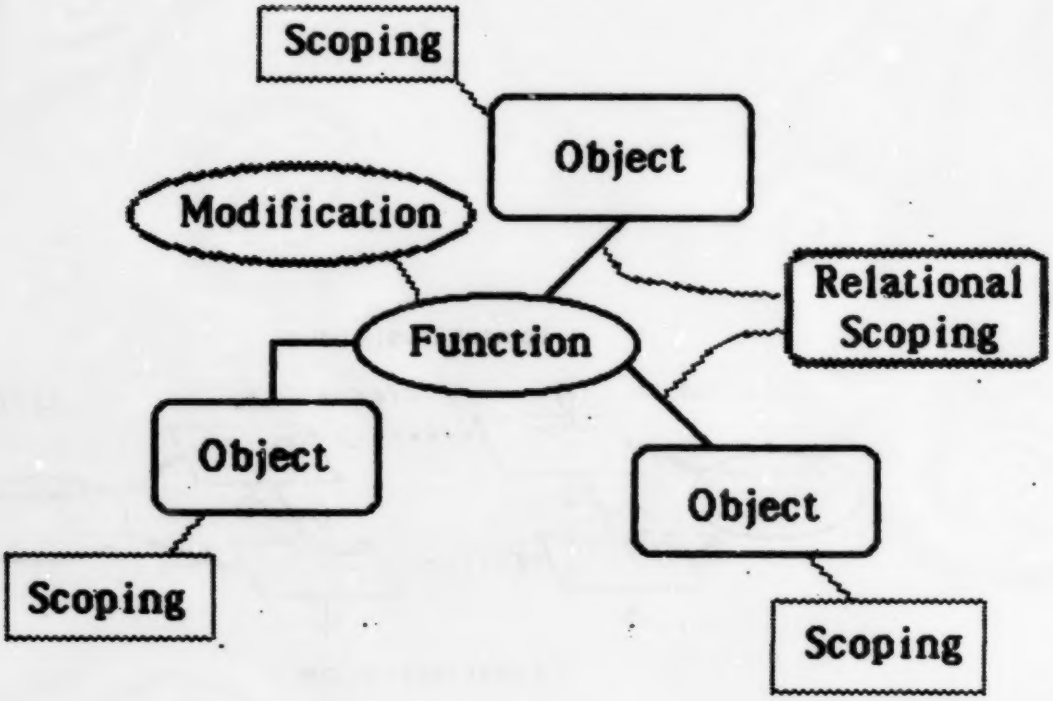

Scoping
Object
Modification
Relational
Scoping
Function
Object
Object
Scoping
Scoping

*Figure 7*

**Select ?? from ?? where??**

1. Default query from data base. This is the most common form of query

**Select coordinates from ?? where ??**

2. Entering "coordinates" confirms "Select" and requires that "from" be answered

**Select coordinates from highway where??**

3. Entering "highway" satisfies the object of "Select", but leaves optional the "where" mapping phrase

**Select coordinates from highway where class = "AA"**

4. Entering "class = AA" completes the requirements for the query. Optional further items can be added before the user commits the query.

**Select coordinates from highway where class = "AA" and condition = "good";**

5. The query is committed by a final semicolon.

*Figure 8*

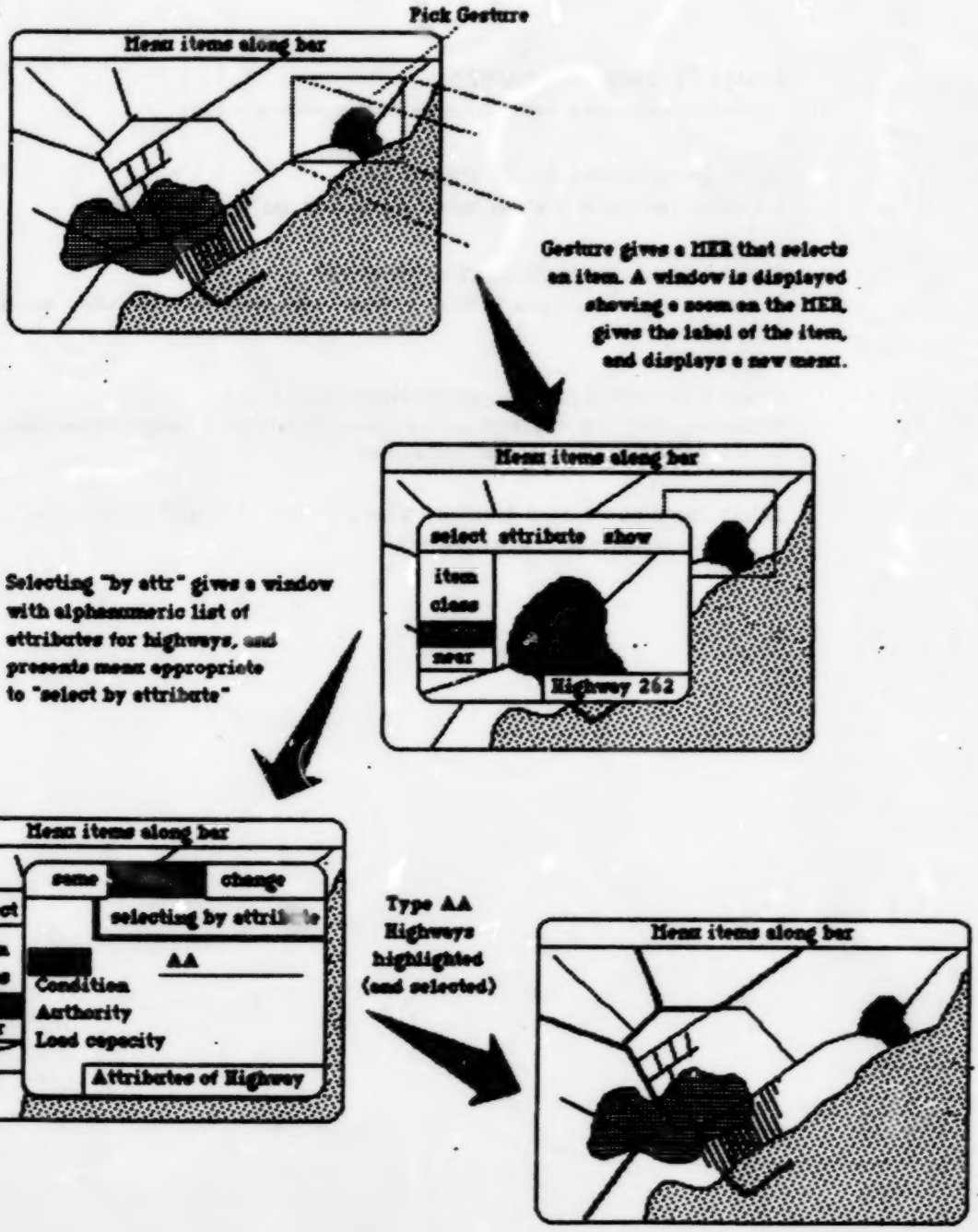

Pick Gesture
Menu items along bar
Gesture gives a MER that selects an item. A window is displayed showing a zoom on the MER, gives the label of the item, and displays a new menu.
Menu items along bar
select attribute show
item class
zoom
Highway 262
Selecting "by attr" gives a window with alphanumeric list of attributes for highways, and presents menu appropriate to "select by attribute"
Menu items along bar
zoom
change
select
selecting by attribute
item class
AA
Condition
Authority
Load capacity
zoom
Attributes of Highway
Type AA
Highways highlighted (and selected)
Menu items along bar

*Figure 9*

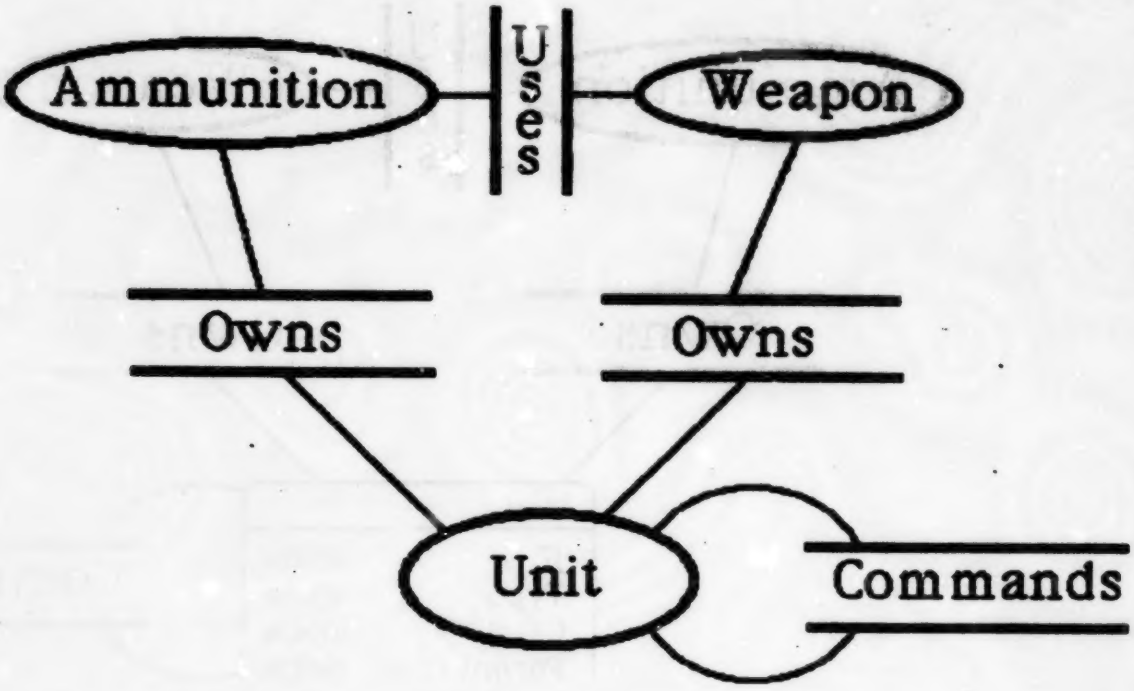



*Figure 10*

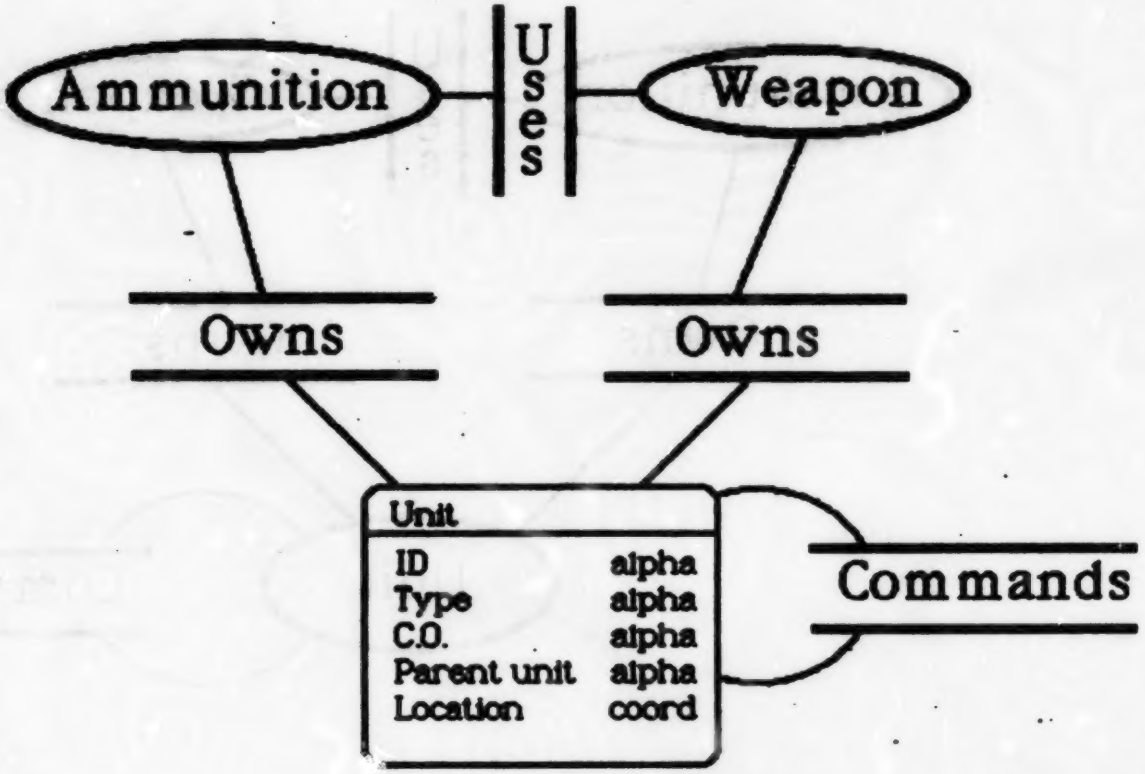

Ammunition
Uses
Weapon
Owns
Owns
Unit
ID alpha
Type alpha
C.O. alpha
Parent unit alpha
Location coord
Commands

*Figure 11*

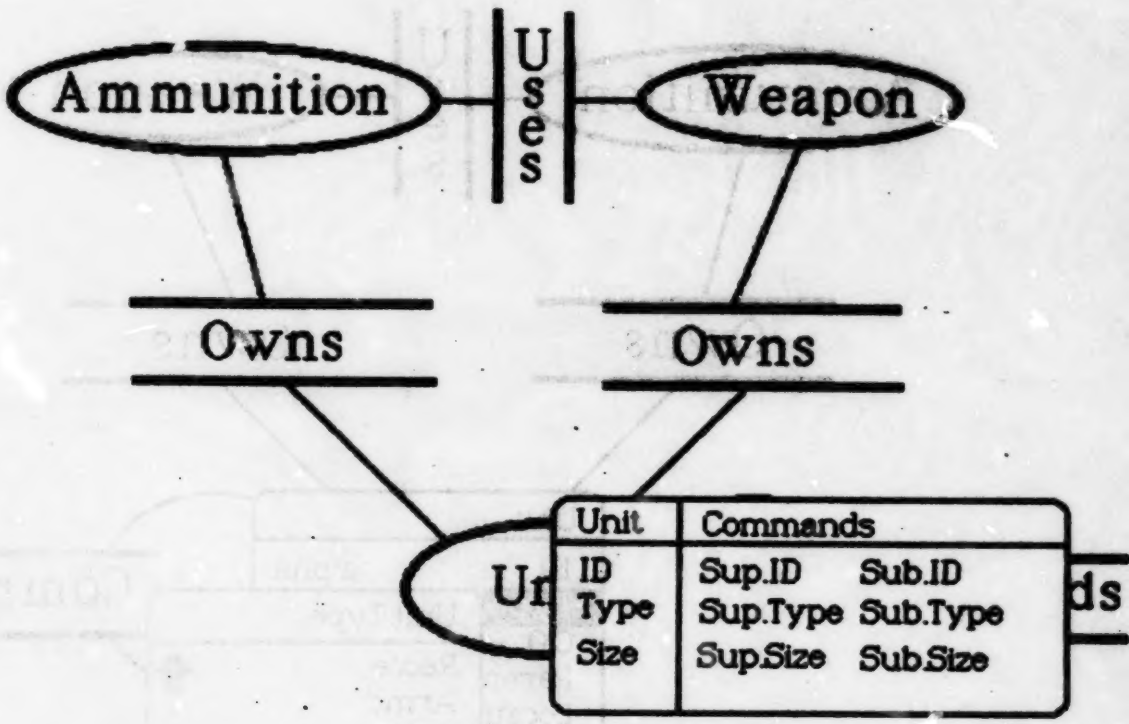

Ammunition
Uses
Weapon
Owns
Owns
Unit
Commands
ID
Type
Size
Sup.ID
Sup.Type
Sup.Size
Sub.ID
Sub.Type
Sub.Size

*Figure 12*

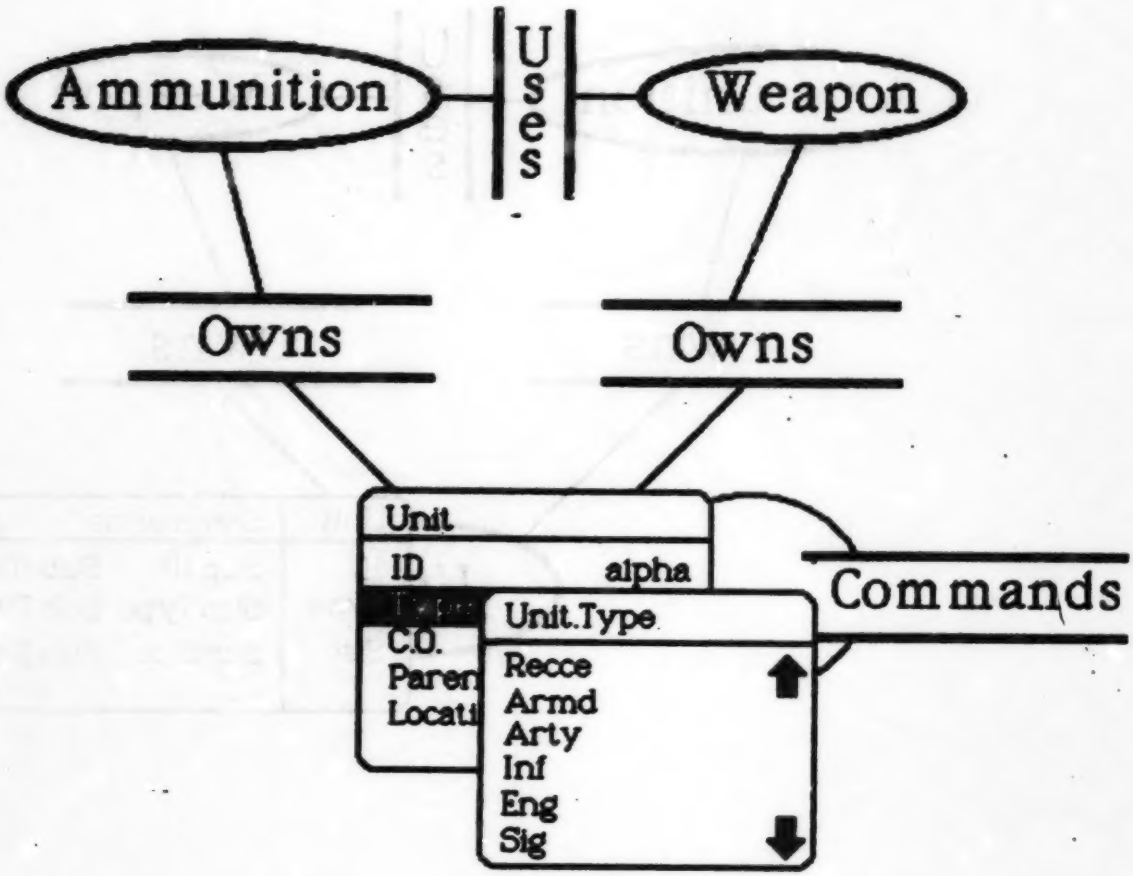

Ammunition
Uses
Weapon
Owns
Owns
Unit
ID
alpha
C.O.
Unit.Type
Recce
Armd
Arty
Inf
Eng
Sig
Commands

*Figure 13*

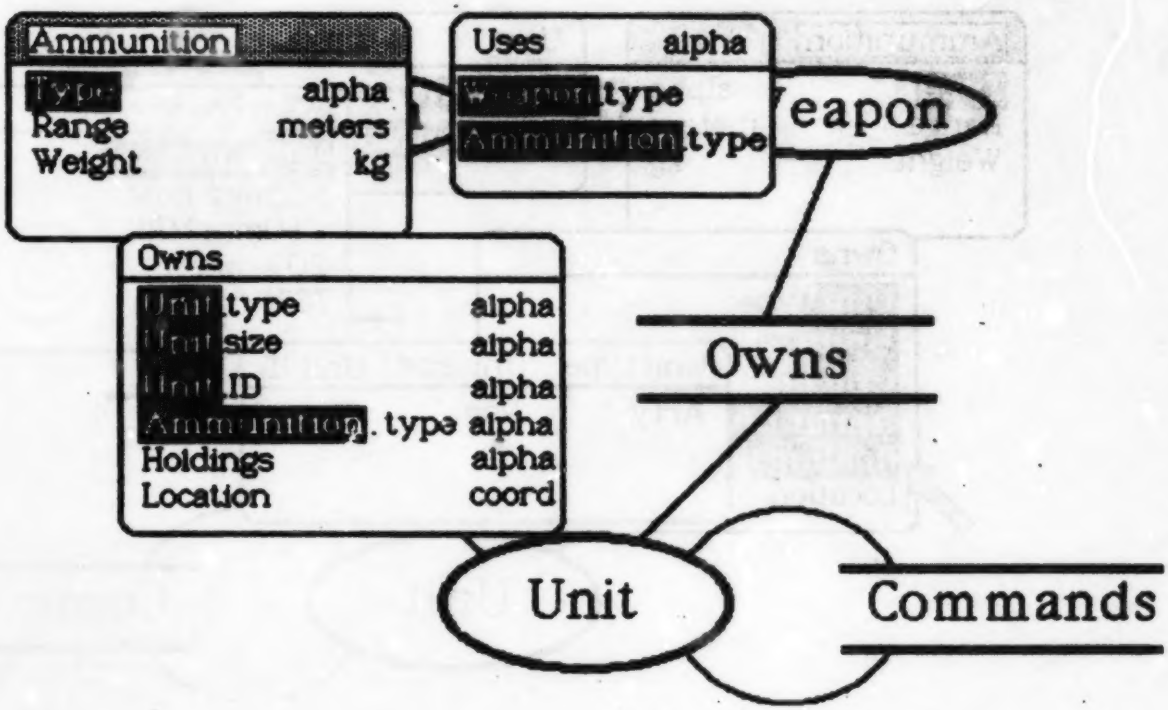

Ammunition
Type alpha
Range meters
Weight kg
Uses alpha
type
type
eapon
Owns
type alpha
size alpha
ID alpha
. type alpha
Holdings alpha
Location coord
Owns
Unit
Commands

*Figure 14*

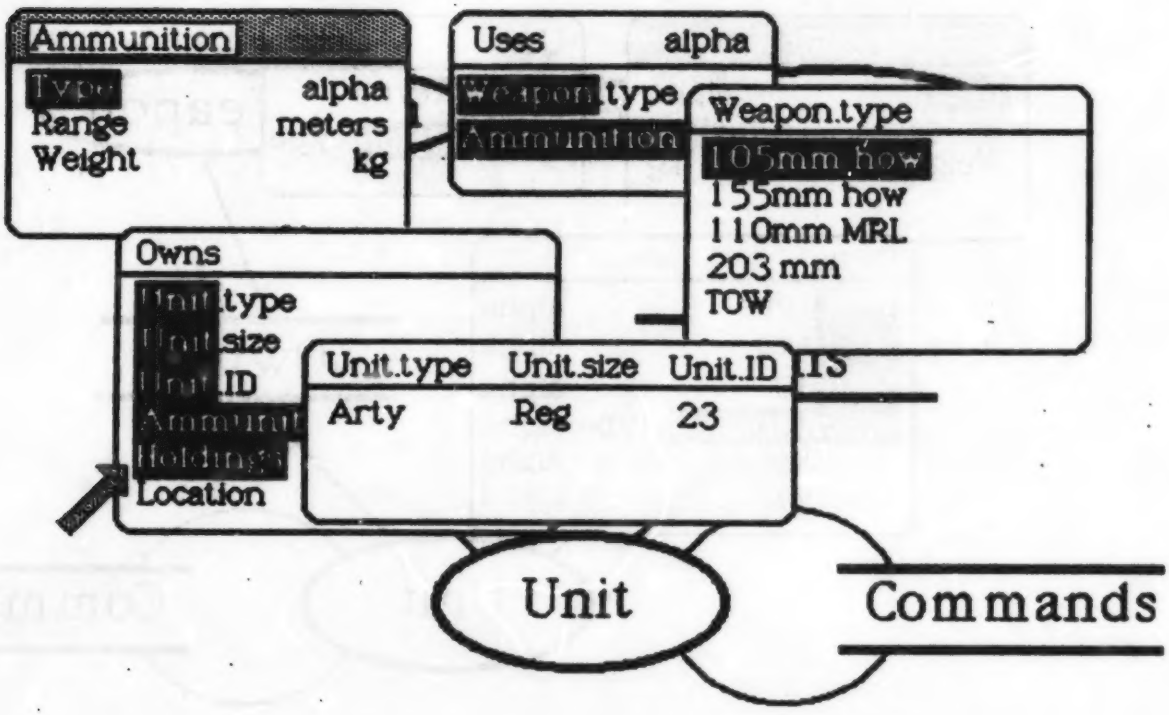

Ammunition
Type alpha
Range meters
Weight kg
Uses alpha
Weapon.type
Ammunition
Weapon.type
105mm how
155mm how
110mm MRL
203 mm
TOW
Owns
Unit.type
Unit.size
Unit.ID
Ammunit
Holdings
Location
Unit.type Unit.size Unit.ID
Arty Reg 23
Unit
Commands

*Figure 15*

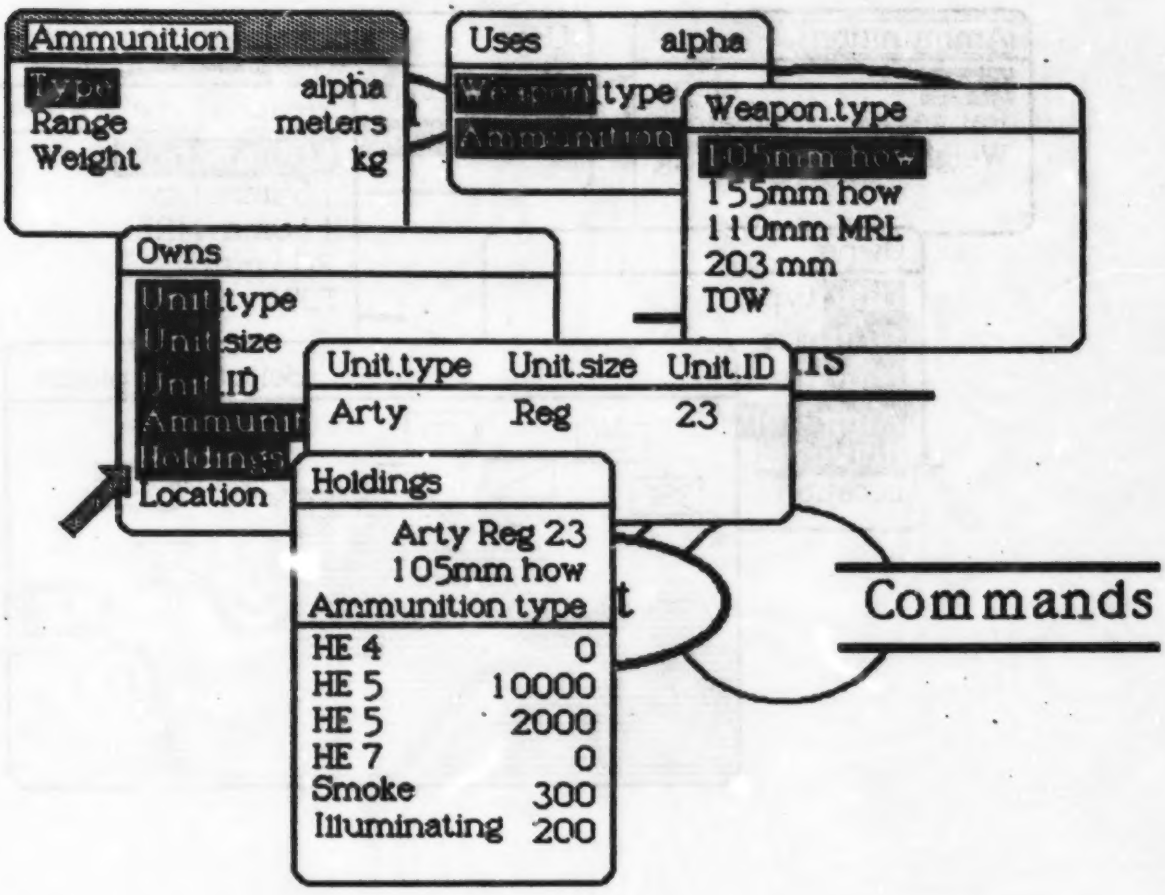

Ammunition
Type alpha
Range meters
Weight kg
Uses alpha
Weapon.type
Ammunition
Weapon.type
105mm how
155mm how
110mm MRL
203 mm
TOW
Owns
Unit.type
Unit.size
Unit.ID
Ammunit
Holdings
Location
Unit.type Unit.size Unit.ID
Arty Reg 23
Holdings
Arty Reg 23
105mm how
Ammunition type
HE 4 0
HE 5 10000
HE 5 2000
HE 7 0
Smoke 300
Illuminating 200
Commands

*Figure 16*

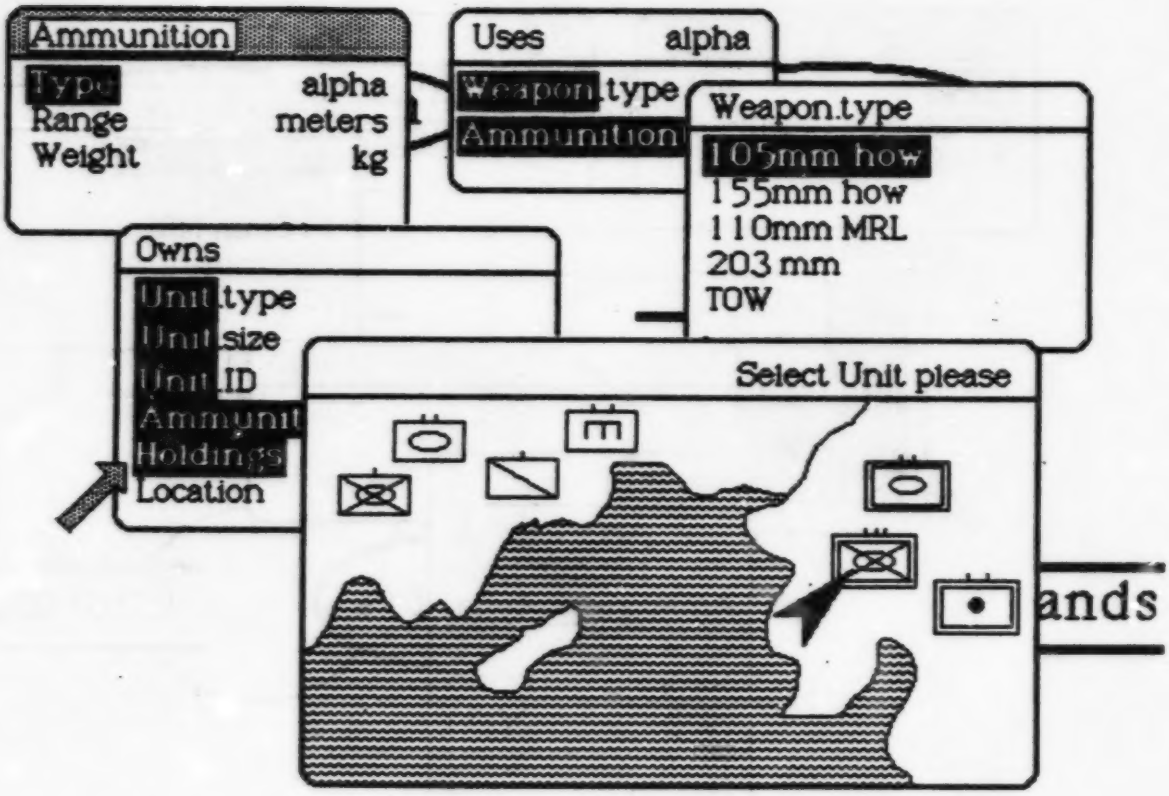

Ammunition
Type alpha
Range meters
Weight kg
Uses alpha
Weapon.type
Ammunition
Weapon.type
105mm how
155mm how
110mm MRL
203 mm
TOW
Owns
Unit.type
Unit.size
Unit.ID
Ammunit
Holdings
Location
Select Unit please
ands

*Figure 17*

Defence and Civil Institute of Environmental Medicine
1133 Sheppard Avenue West, Post Office Box 2000
Downsview, Ontario, Canada, Telephone (416) 633-4240